Real-Time Anomaly Segmentation for Road Scenes

Andrea Delli
Politecnico di Torino
s331998
s331998@studenti.polito.it

Christian Dellisanti
Politecnico di Torino
s306027
s306027@studenti.polito.it

Giorgia Modi
Politecnico di Torino
s330519
s330519@studenti.polito.it

Abstract

This study addresses real-time anomaly segmentation in road scenes, a critical task for applications like autonomous driving. We evaluate baseline segmentation models (ENet, ERFNet, BiSeNet) pretrained on Cityscapes, with different metrics and methods, incorporating enhancements such as temperature scaling, void classifiers, and fine-tuning with different loss functions. Performance is assessed on benchmark datasets using AuPRC, FPR95, and mIoU metrics. Results highlight MaxLogit's robustness, BiSeNet's efficiency, and the benefits of calibration and task-specific loss functions for anomaly detection. These findings offer insights into building efficient, reliable systems for real-world environments. The source code of this project is available at https://github.com/RonPlusSign/AnomalySegmentation.

Figure 1. Visual comparison of anomaly segmentation output of MSP and MaxLogit , applied on an image of the Road Anomaly dataset. The image on the top left shows the input from the dataset, followed by the ground truth segmentation. The image on the bottom left the MSP output and at the bottom right the MaxLogit .

1. Introduction

Anomaly segmentation is a critical task in computer vision that involves identifying regions within an image that deviate from expected patterns. This capability has significant real-world applications, including detecting road obstacles for autonomous driving vehicles or identifying defective objects in industrial systems. Anomalies often represent unpredictable or rare events, such as fallen debris on a roadway, making their detection essential for safety and operational efficiency.

In this context, *per-pixel anomaly segmentation* focuses on the identification of anomalous regions at the pixel level. This task is particularly challenging due to the need to distinguish between In-Distribution (ID) and Out-of-Distribution (OOD) samples that the model has not encountered during training. In the domain of autonomous driving, per-pixel anomaly segmentation ensures that the system can accurately localize and respond to hazards, even when faced with novel or unexpected scenarios.

Achieving *real-time performance* for per-pixel anomaly segmentation is vital for its deployment in safety-critical applications, such as autonomous vehicles, which must process sensor data with minimal latency to make fast and reliable decisions. This requires methods that strike a balance between computational efficiency and high detection accuracy.

In this paper, we explore per-pixel anomaly segmentation through a series of experiments designed to evaluate and enhance its performance. We begin by establishing and testing three baseline models for image segmentation (ENet [13], ERFNet [15], and BiSeNet [18]), pretrained on Cityscapes [4] and tested on different task-specific datasets with different methods, followed by the application of temperature scaling for confidence calibration. Additionally, we introduce a *Void Classifier* [5] to explicitly leverage OOD knowledge, and lastly we analyze the effects of different training loss functions specifically designed for OOD detection.

Starting with pre-trained image segmentation models as baselines, we aim to improve their performance through task-specific enhancements, including confidence calibration and optimized training loss functions, ultimately providing insights into designing robust and efficient anomaly detection systems for real-world road scenes.

2. Related works

Real-time semantic segmentation requires a careful balance between computational efficiency and the ability to capture both spatial and contextual information. In this section, we discuss three key architectures designed with this objective: ENet, ERFNet, and BiSeNet.

ENet (Efficient Neural Network) ENet [13] is a lightweight encoder-decoder network designed for resource-constrained environments. The encoder is optimized to compress spatial information early using a combination of downsampling and low-dimensional feature representations. Key innovations include the use of *bottleneck modules*, which reduce feature dimensionality via 1×1 projections, followed by either regular, dilated, or asymmetric convolutions (Fig. 2). The decoder, in contrast, is minimalist, focusing solely on upsampling the encoder's compressed representations to produce pixel-level predictions. ENet's design emphasizes computational efficiency by limiting the size of the decoder, making ENet suitable for real-time applications on embedded devices.

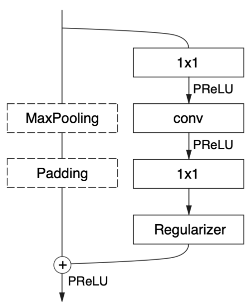

Figure 2. ENet bottleneck module.

ERFNet (Efficient Residual Factorized Network) ERFNet [15] is a deep neural network designed for real-time semantic segmentation. It builds on ENet by retaining its encoder-decoder structure and minimal decoder, while introducing the *non-bottleneck-1D* module to improve efficiency of the residual layer (Fig. 3). These modules use factorized 1D convolutions, significantly reducing computational costs and number of parameters compared to traditional 2D convolutions while maintaining high representational power. The encoder combines these modules with downsampling layers and dilated convolutions to extract multi-scale features, while the lightweight decoder focuses on upsampling using transposed convolutions to recover spatial resolution. This architecture achieves a strong balance between accuracy and speed, making ERFNet ideal for real-time applications.

BiSeNet (Bilateral Segmentation Network) BiSeNet [18] introduces a dual-path architecture to address the trade-off between preserving spatial details and capturing a large receptive field (Fig. 4). The *Spatial Path* (SP) uses three convolutional layers with a stride of 2 to retain high spatial resolution. Concurrently, the *Context Path* (CP), built on lightweight backbones such as Xception, captures contextual information by aggressively downsampling feature maps and applying global average pooling. To fuse the complementary outputs of SP and CP, BiSeNet employs a *Feature Fusion Module* (FFM), which selectively combines features from both paths. Additionally, an *Attention Refinement Module* (ARM) enhances feature representations by focusing on relevant spatial and contextual regions. This architecture ensures both high-resolution segmentation and computational efficiency.

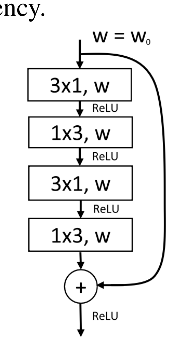

Figure 3. ERFNet non-bottleneck-1D module.

3. Methods

Various methods have been proposed for identifying anomalous samples, leveraging the model's predictive behavior and internal representations. Here, we discuss five key methods: Maximum Softmax Probability (MSP), Maximum Logit (MaxLogit), Maximum Entropy (MaxEntropy), Void Classifier and Mahalanobis Distance.

All methods provide pixel-wise anomaly scores $s(x) \in \mathbb{R}^{|\mathcal{Z}|}$, where $x \in \mathcal{X}$ represents an image. Anomalies correspond to higher values of $s(x)$. $\mathcal{Z}$ denotes the set of image coordinates, and $\mathcal{X}$ is a dataset with N images. $\sigma(\cdot)$ denotes the softmax function and $f_z^c(x)$ represents the logits for class c.

Maximum Softmax Probability (MSP) [8] The Maximum Softmax Probability (MSP) evaluates the confidence of the semantic segmentation model based on the softmax probabilities. The anomaly score for each pixel $z \in \mathcal{Z}$ is computed as:

$$s_z(x) = 1 - \max_{c \in \mathcal{C}} \sigma(f_z^c(x)). \tag{1}$$

MSP assumes that the highest softmax probability corresponds to the model's confidence. Low confidence (i.e., low maximum probability) is interpreted as a sign of an anomaly.

While simple and computationally efficient, MSP may fail in cases where softmax probabilities are overconfident.

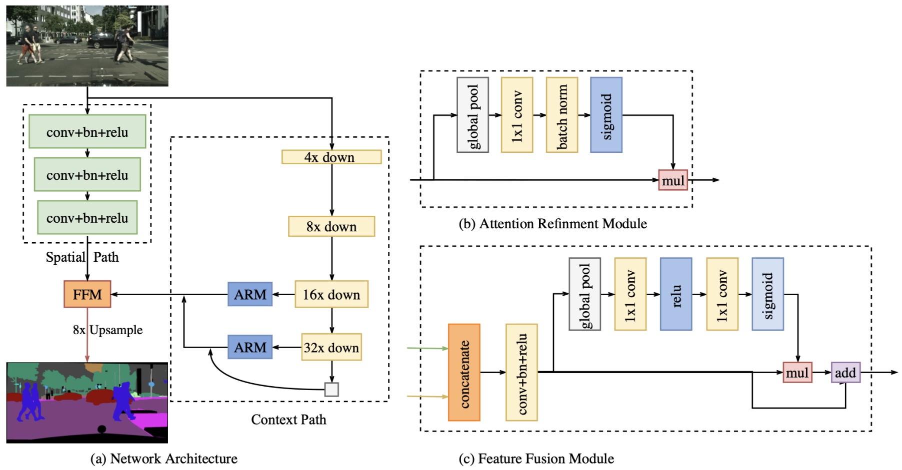

Figure 4. BiSeNet's dual-path architecture: Spatial Path (SP), Context Path (CP), and Feature Fusion Module (FFM).

To address this, *temperature scaling* [10] can be applied as a post-processing technique to calibrate the classifier's confidence in predictions. A classifier is considered calibrated when its predicted probabilities match the actual frequency of correct predictions. The MSP anomaly score with temperature scaling is computed for each pixel $z \in \mathcal{Z}$ as:

$$s_z(x) = 1 - \max_{c \in \mathcal{C}} \sigma(f_z^c(x)/T), \tag{2}$$

where T represents the temperature value.

Maximum Logit (MaxLogit) [7] The Maximum Logit (MaxLogit) method operates on the logits directly rather than relying on softmax probabilities. The anomaly score is defined as:

$$s_z(x) = -\max_{c \in \mathcal{C}} f_z^c(x). \tag{3}$$

By avoiding softmax, MaxLogit prevents inflated confidence scores, offering a computationally efficient and effective measure of anomaly likelihood.

Maximized Entropy (MaxEntropy) [3] MaxEntropy is based on the entropy of softmax outputs. The anomaly score is computed as:

$$s_z(x) = -\sum_{c \in \mathcal{C}} \sigma(f_z^c(x)) \log\left(\sigma(f_z^c(x))\right). \tag{4}$$

Entropy measures the uncertainty of predictions, with high entropy values indicating that the model is unsure about the class assignment, which aligns with the behavior expected for OOD pixels. MaxEntropy provides more nuanced uncertainty estimates compared to MSP but is computationally more intensive.

Void Classifier In this approach, a deep neural network $f : \mathcal{X} \mapsto \mathbb{R}^{|\mathcal{Z}| \times (|\mathcal{C}|+1)}$ is trained with an additional "void" class to model anomalies. The anomaly score for each pixel $z \in \mathcal{Z}$ is given by the softmax score for the void class:

$$s_z(x) = \sigma(f_z^{\text{void}}(x)), \quad x \in \mathcal{X}. \tag{5}$$

By explicitly modeling anomalies as the void class, this method reduces reliance on post-hoc anomaly detection and can be integrated into existing segmentation pipelines. Its effectiveness depends on accurate void class annotation during training.

Mahalanobis Distance [9] The Mahalanobis Distance calculates the distance of a pixel's feature representation to the nearest class distribution in the feature space. Assuming that the features $h_z^{L-1}(x)$ of the penultimate layer follow a Gaussian distribution, the anomaly score is given by:

$$s_z(x) = \min_{c \in \mathcal{C}} \left(h_z^{L-1}(x) - \hat{\mu}_c\right)^T \hat{\Sigma}^{-1} \left(h_z^{L-1}(x) - \hat{\mu}_c\right), \tag{6}$$

where $\hat{\mu}_c$ and $\hat{\Sigma}$ represent the mean of the features for class c and the tied covariance, respectively. The Mahalanobis distance captures the likelihood of the pixel belonging to a known class, with higher values indicating anomalies.

3.1. Loss Functions

In this work, we explore the application of several loss functions for fine-tuning a pre-trained semantic segmentation model. Below, we provide an overview of each loss function.

Cross-Entropy Loss The *cross-entropy loss* is a standard objective for classification tasks and is adapted here for semantic segmentation. For a single pixel at position $z \in \mathcal{Z}$, with predicted logits $f_z^c(x) \in \mathbb{R}^C$ and ground truth label y_z, the loss is:

$$\mathcal{L}_{\text{CE}} = -\log\left(\sigma(f_z^{y_z}(x))\right), \tag{7}$$

where $\sigma(f_z^{y_z}(x)) = \frac{\exp(f_z^{y_z}(x))}{\sum_{c=1}^{C}\exp(f_z^c(x))}$.

Focal Loss The *focal loss* [6] extends the cross-entropy loss by introducing a modulating factor to emphasize harder examples. For a single pixel $z \in \mathcal{Z}$, the loss is:

$$\mathcal{L}_{\text{FocalLoss},z} = -\alpha(1-\sigma(f_z^{y_z}(x)))^{\gamma}\log(\sigma(f_z^{y_z}(x))), \tag{8}$$

where $\alpha \in [0,1]$ and $\gamma \geq 0$ control the weighting of examples and the focusing effect, respectively.

Logit Normalization The *Logit Normalization (LogitNorm)* technique [17] aims to mitigate the issue of overconfidence in deep learning models, which often produce highly confident predictions even for out-of-distribution (OOD) inputs. LogitNorm modifies the traditional cross-entropy loss by normalizing the logits, i.e., the pre-softmax outputs of the model, to have a constant norm. The normalized logits are defined as:

$$\hat{f}_z(x) = \frac{f_z(x)}{\|f_z(x)\|}. \tag{9}$$

The cross-entropy loss is then computed as:

$$\mathcal{L}_{\text{CE},z} = -\log(\sigma(\frac{\hat{f}_z^{y_z}(x)}{T})), \tag{10}$$

where T is a scaling factor that modulates the magnitude of the logit vector after normalization. This technique reduces overconfidence by normalizing logits to have a constant norm, making the model less sensitive to large logit values.

Enhanced Isotropy Maximization Loss The *Enhanced Isotropy Maximization Loss (IsoMax+)* [12] extends the IsoMax loss to improve out-of-distribution (OOD) detection by replacing the traditional linear transformation with a distance-based approach. The logit for class c is defined as the negative scaled Euclidean distance between the normalized feature vector $\hat{\mathbf{f}}_z$ from the penultimate layer and the normalized class prototype $\hat{\mathbf{p}}_c$:

$$f_z^c = -|d_s| \cdot \|\hat{\mathbf{f}}_z - \hat{\mathbf{p}}_c\|, \tag{11}$$

where $\hat{\mathbf{f}}_z$ is the normalized feature vector, $\hat{\mathbf{p}}_c$ is the normalized class prototype, and $|d_s|$ is a learnable scalar that controls the scaling of distances.

The IsoMax+ loss function is then defined as:

$$\mathcal{L}_{\text{IsoMax+},z} = -\log\left(\sigma(E_s \cdot f_z^{y_z})\right), \tag{12}$$

where E_s represents the entropic scale. IsoMax+ enhances isotropy in the feature space, bringing in-distribution samples closer to their class prototypes while pushing OOD samples away. It improves OOD detection without additional hyperparameter tuning or external datasets. For ERFNet, which outputs a $C \times H \times W$ tensor, the method is adapted by reshaping and normalizing the features ($B \times C \times H \times W$) to compute distances between pixel-level feature vectors and class prototypes, ensuring compatibility with ERFNet's output while preserving IsoMax+ properties.

4. Experiments

In our experiments, we evaluate an anomaly segmentation framework using inference methods like Maximum Softmax Probability (MSP), Maximum Logit (MaxLogit), Maximum Entropy (MaxEntropy), and Mahalanobis distance. We also explore the effects of Temperature Scaling for calibration and the void class from the Cityscapes dataset on anomaly detection. Additionally, we test various loss functions, including Focal Loss, Cross-Entropy Loss, IsoMax+, and LogitNorm, to enhance performance.

Experiments are conducted on benchmark datasets such as RoadAnomaly, Fishyscapes, and SegmentMelfYouCan, using pre-trained models like ERFNet, ENet, and BiSeNetV1.

4.1. Metrics

We evaluate performance using three metrics: Area under the Precision-Recall Curve (AuPRC), False Positive Rate at 95% True Positive Rate (FPR95), and Mean Intersection over Union (mIoU).

The **AuPRC** metric quantifies how well the anomaly scores separate anomalies from non-anomalies. The metric is threshold-independent, meaning it considers how precision and recall change as the threshold for classifying points as anomalies is varied. A higher AuPRC indicates better separation between anomalies and non-anomalies, with a value closer to 1 being ideal.

The **FPR95** metric evaluates the false positive rate when the true positive rate is 95%. The positive class is defined as "anomaly", and false positives are non-anomaly pixels incorrectly predicted as anomalies.

The **mIoU** metric quantifies the overlap between predicted and ground truth segmentation masks. It is defined as:

$$\text{mIoU} = \frac{1}{C}\sum_{c=1}^{C}\frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c + \text{FN}_c}, \tag{13}$$

where C is the number of classes, and TP_c, FP_c, and FN_c are the true positive, false positive, and false negative pixels for class c, respectively. The mIoU values reported in the tables of this paper (1, 2, 3, 4) refer to a mIoU evaluation over the Cityscapes dataset [4].

4.2. Datasets and Benchmarks

Here, we describe the main datasets and benchmarks relevant to our work. In our experiments we have only used the validation splits of these benchmark datasets, which are publicly available for download but contain a limited number of different road surfaces and diverse obstacle types than the whole dataset benchmarks.

Cityscapes [4] is a widely-used dataset for semantic segmentation in urban driving scenarios. It consists of 5,000 high-resolution images with dense pixel-level annotations across 19 semantic classes, captured in diverse European cities. Cityscapes provides a strong foundation for segmentation models, particularly for road scene understanding.

Fishyscapes [1] is a benchmark designed to evaluate anomaly detection in semantic segmentation. It includes three datasets, but only two were used in our work: FS Static and FS Lost and Found. *FS Static* is based on the Cityscapes validation set and is divided into a public validation set of 30 images with 30 OOD objects overlaid, and a hidden test set of 1000 images. *FS Lost and Found* is derived from the Lost and Found dataset [14] and consists of 100 validation images and 275 test images with pixel-level annotations of small, anomalous objects on the road.

RoadAnomaly [11] focuses on detecting anomalies in real-world road scenes. This dataset features 60 images with pixel-level annotations with various anomalous objects, such as animals or atypical vehicles, appearing in unpredictable locations within the image, making it a challenging test for anomaly segmentation models.

SegmentMeIfYouCan [2] is a benchmark for anomaly segmentation that introduces two key datasets: *RoadAnomaly21* and *RoadObstacle21*. *RoadAnomaly21* contains 100 real-world test images and 10 validation images where anomalies can appear anywhere, emphasizing general anomaly detection. *RoadObstacle21* consists of 327 test images and 30 validation images, restricting the region of interest to the drivable road area, and focuses on the detection of potential hazards such as fallen objects, with annotations tailored for obstacle segmentation tasks. In both datasets, the pixel-level annotations include three classes: 1) anomaly / obstacle, 2) not anomaly / not obstacle, and 3) void.

4.3. Implementation details

In this section, we detail our implementation protocol for each model. We used the pre-trained versions of ERFNet[1], ENet[2], and BiSeNetV1[3] available in their official GitHub repositories. All models were pre-trained on the Cityscapes dataset with 19 semantic classes.

For the void classifier experiment (Section 4.6), we fine-tuned the models for 20 epochs to include the void class in the anomaly score by freezing all layers except the final one. Similarly, for the additional losses experiment (Section 4.7), we fine-tuned ERFNet under the same conditions using different loss functions.

We pre-processed the data by resizing each image to 512×1024 pixels. The data augmentations and hyperparameters were adopted directly from the original papers to ensure consistency with the authors' implementations. Below, we detail the specific settings and configurations used for each model.

For the ERFNet model, we used the Adam optimizer with an initial learning rate of 5×10^{-5} and a weight decay of 10^{-4}. The learning rate was adjusted using a LambdaLR scheduler based on Equation 14. To study the effect of various losses and OOD methods, we experimented with Focal Loss ($\gamma = 0$, $\alpha = 1$), Cross Entropy Loss, IsoMax+ ($E_s = 10$), and LogitNorm (without temperature scaling).

For the ENet model, we used the Adam optimizer with an initial learning rate of 5×10^{-5} and a weight decay of 0.0002. We adopted a step learning rate scheduler with a decay by $\gamma = 0.1$ every 7 epochs. The loss function used for ENet was the Cross Entropy Loss. To account for class imbalances in the Cityscapes dataset, we calculated dataset weights using the function described in the original ENet paper and provided them as input to the Cross Entropy Loss. For ERFNet, we used the Cityscapes dataset weights provided in the official GitHub repository.

For the BiSeNetV1 model, we used the SGD optimizer with an initial learning rate of 2.5×10^{-3}, a momentum of 0.9, and a weight decay of 10^{-4}. The learning rate was scheduled using the LambdaLR approach, as defined in Equation 14. The loss function is composed of three separate terms: one for the main output and two for the auxiliary outputs of the BiSeNetV1 architecture, all based on Ohem Cross Entropy loss [16] with a threshold of 0.7, in line with the original implementation.

The Lambda LR scheduler is defined as:

$$\lambda(t) = \left(1 - \frac{t-1}{T}\right)^{0.9} \tag{14}$$

where t is the epoch and T is the total number of epochs.

4.4. Baselines

To compare the effect of different methods (MSP, MaxLogit, MaxEntropy and Mahalanobis) for anomaly segmentation, we evaluated the performance of an ERFNet model pre-trained with 19 Cityscapes classes on different datasets: RoadAnomaly, RoadAnomaly21, RoadObstacle21, Fishyscapes Static, Fishyscapes Static Lost and Found. In particular, to compute the Mahalanobis method we have implemented an initial calculation of the mean and

[1] https://github.com/Eromera/erfnet_pytorch
[2] https://github.com/davidtvs/PyTorch-ENet
[3] https://github.com/CoinCheung/BiSeNet

Table 1. Performance results of ERFNet across various benchmark datasets for different metrics used as baselines. The table reports mIoU (higher is better), AuPRC (higher is better), and FPR95 (lower is better) metrics, evaluated for different methods: MSP, MaxLogit, Max Entropy, Mahalanobis. Results are evaluated on SMIYC RA-21, SMIYC RO-21, FS L&F, FS Static, and Road Anomaly datasets, and the best performance for each dataset and metric is highlighted in **bold black**.

Method	Cityscapes	SMIYC RA-21		SMIYC RO-21		FS L&F		FS Static		Road Anomaly	
	mIoU ↑	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓
MSP	72.20	29.10	62.51	2.71	64.97	1.75	50.76	7.47	41.82	12.43	82.49
MaxLogit	72.20	**38.32**	**59.34**	4.63	**48.44**	**3.30**	**45.49**	**9.50**	40.30	**15.58**	**73.25**
Max Entropy	72.20	31.01	62.59	3.05	65.60	2.58	50.37	8.83	41.52	12.68	82.63
Mahalanobis	72.20	30.88	74.49	**9.64**	52.43	2.94	55.23	8.93	**39.34**	13.53	79.63

tied covariance matrix of each output of the ERFNet model on the Cityscapes dataset.

The performance results reported in Table 1 show that the MaxLogit method generally outperforms all other methods, because it can effectively distinguish between classes that are very similar. In contrast, the MSP method, which relies on Softmax, tends to reduce the logits, making the differences between classes less pronounced and harder to differentiate.

The second best method is Mahalanobis, which outperforms the MSP method because it provides a structured way to detect anomalies, even though is constrained by the Gaussian assumption. Despite this, the computational cost of Mahalanobis is not negligible; it requires the estimation and inversion of covariance matrices, which can be expensive, especially for high-dimensional feature spaces.

Overall, the best method is MaxLogit for both its simplicity and performance. Figure 5 provides a qualitative comparison of the anomaly detection performance of the different methods on an image from the Road Anomaly dataset.

4.5. Temperature scaling

In this experiment, we implemented Temperature Scaling in the MSP method using Equation 2 and we studied the effect of different temperature values on benchmark datasets. We have had trials with different temperatures in a range between 0.5 and 2.0. The results reported in Table 2 show that MSP with a temperature of 1.85 outperforms all other temperature values, indicating that the network is overconfident in the results and requires calibration.

Figure 6 provides a qualitative comparison of the anomaly detection performance of different temperature values on an image from the Road Anomaly dataset.

4.6. Void classifier

In this experiment, we fine-tuned ERFNet, ENet and BiSeNetV1 by enabling the *void* output channel, representing the 20th class in the Cityscapes dataset, typically used for background or unannotated areas. We reinterpreted it as an anomaly class, encompassing all elements outside of the 19 predefined Cityscapes categories.

The three baseline models were pre-trained on Cityscapes with the cross-entropy loss, ignoring the void class. To adapt them, we fine-tuned for 20 epochs on the Cityscapes dataset setting the weight of the void class to 1, in order to approximate an anomaly distribution using the dataset's void regions directly during training, obtaining the *Void Classifier* [2] defined in section 3. During inference, anomaly detection was performed by isolating the void output class and treating it as an anomaly score.

Table 3 presents the performance of networks trained as Void Classifiers, tested on various benchmark datasets, with Figure 7 showcasing a qualitative comparison on an image from the Road Anomaly dataset.

BiSeNet consistently achieves the highest AuPRC scores across all datasets, notably outperforming others on SMIYC RA-21 (46.79%) and FS Static (42.52%), demonstrating superior precision and recall in detecting void classes.

On the other hand, ERFNet achieves the best FPR95 results on most datasets, excelling in FS L&F (13.17%), reflecting its robustness in minimizing false positives.

Comparing ERFNet trained as a Void Classifier (first row of Table 3) with baseline methods (Table 1), the Void Classifier generally performs worse, except for improvements in FS L&F (FPR95) and FS Static (AuPRC). Also the mIoU is slightly lower, suggesting that the model may benefit from further fine-tuning.

4.7. Effect of Training Loss function

In this experiment, we fine-tuned the pre-trained ERFNet model with various loss functions to evaluate their impact on training dynamics. We conducted six experiments, comparing individual and combined effects of common losses (cross-entropy and focal loss) and outlier detection-specific techniques (logit normalization and IsoMax+). First, as a baseline, the pre-trained ERFNet model was evaluated using cross-entropy loss without modifications. Next, the model was fine-tuned with focal loss alone to assess its standalone performance. Finally, we explored combinations of logit normalization or IsoMax+ with cross-entropy or focal loss to evaluate their synergistic effects. The focal loss was

Table 2. Performance results of ERFNet across various benchmark datasets for the MSP method with various temperatures.

Method	Cityscapes	SMIYC RA-21		SMIYC RO-21		FS L&F		FS Static		Road Anomaly	
	mIoU ↑	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓
MSP	72.20	29.10	62.51	2.71	64.97	1.75	50.76	7.47	41.82	12.43	82.49
MSP ($t = 0.5$)	72.20	27.06	62.73	2.42	**63.23**	1.28	66.74	6.60	43.48	12.19	**82.02**
MSP ($t = 0.75$)	72.20	28.16	**62.48**	2.57	64.05	1.49	51.85	6.99	42.49	12.32	82.28
MSP ($t = 1.1$)	72.20	29.41	62.59	2.77	65.52	1.86	50.39	7.69	41.59	12.47	82.62
MSP ($t = 1.85$)	72.20	**30.60**	64.72	**3.01**	70.41	**2.57**	**48.65**	**9.26**	**40.98**	**12.65**	84.14

Table 3. Performance results across datasets of different networks trained as Void Classifiers.

Network	Cityscapes	SMIYC RA-21		SMIYC RO-21		FS L&F		FS Static		Road Anomaly	
	mIoU ↑	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓
ERFNet	71.94	20.96	**70.65**	0.95	99.70	11.95	**13.17**	19.09	**54.49**	9.62	**89.60**
ENet	34.48	12.82	96.94	0.66	99.80	2.27	56.55	11.04	76.59	12.43	91.83
BiSeNet	67.10	**46.79**	80.71	**6.20**	**99.58**	**16.80**	70.48	**42.52**	57.13	**19.54**	93.59

used as a straightforward substitution for cross-entropy loss in these experiments, without requiring additional modifications to the methods described earlier in Section 3.1.

Table 4 reports the results of this experiment. Additionally, Figure 8 provides a qualitative comparison of the anomaly detection performance on an image from Road Anomaly dataset.

Overall, IsoMax+ (IMP) loss and Logit Normalization (LN) show promise in enhancing anomaly detection, but their effectiveness varies depending on the dataset and method.

For both the MSP and MaxEntropy methods, IMP combined with Cross Entropy (CE) achieves the best results. This aligns with the design of IMP, which is inherently compatible with CE due to its cross-entropy-based formulation. Interestingly, incorporating Logit Normalization (LN) with CE or Focal Loss (FL) yields mixed results, with improvements on datasets like FS Static but declines on SMIYC RA-21 and RO-21. These findings suggest that LN's regularization effect is sensitive to the characteristics of the dataset and could disrupt valuable information embedded in raw logits. The combination of IMP with FL generally underperforms compared to FL alone, suggesting that IMP's isotropy-based optimization may conflict with FL's focus on hard-to-classify examples.

The MaxLogit method stands out as the overall best-performing approach. Within MaxLogit, IMP+CE consistently achieves strong results across most datasets and this result reinforces the compatibility between IMP and CE. However, adding LN to either CE or FL in MaxLogit typically degrades performance, likely because MaxLogit relies on raw logits, and LN's normalization constrains their range, potentially reducing effectiveness.

The Mahalanobis method shows strong results with traditional loss functions like CE and FL, particularly for FPR95 metrics. FL stands out as particularly effective, achieving the best overall results across many datasets. However, combining IMP with CE yields poor outcomes, likely due to incompatibilities between Mahalanobis's distance-based anomaly detection and IMP's isotropy-focused design. Once again, the effect of LN is highly variable depending on the dataset.

An interesting observation from Figure 8 is that IsoMax+ tends to produce smoother and more calibrated predictions, suggesting an enhancement in the model's ability to express uncertainty, which is critical for reliable anomaly detection. The lower mIoU values observed for IsoMax+ (27.73 and 17.79) indicate the need for additional fine-tuning due to the introduction of new parameters in the final layer.

While both IMP and LN offer improvements for out-of-distribution detection, their effectiveness varies depending on the method and dataset. IMP+CE is the most consistent and effective combination, while LN's impact varies based on the specific characteristics of the dataset.

4.8. Performance comparison

The performance comparison between the analyzed networks, measured as the average forward time on a T4 GPU using the Cityscapes dataset, highlights the critical role of computational efficiency in *real-time* semantic segmentation. As can be seen in Table 5 BiSeNet demonstrates the shortest forward time (18.16 ms) and thus the highest frames-per-second (55.08), significantly outperforming ERFNet and ENet in processing speed. This metric is crucial because real-time applications such as autonomous driving demand low-latency segmentation to ensure timely and accurate scene understanding. Faster networks like BiSeNet can provide the necessary responsiveness while maintaining segmentation quality, making them more suitable for real-world deployments.

Table 4. Performance results of ERFNet fine-tuned with different combinations of methods and loss functions: Cross Entropy (CE), Focal Loss (FL), Logit Normalization (LN), and IsoMax+ Loss (IMP). The best performance for each dataset and metric is highlighted in **bold black**, while the best overall results across all metrics are highlighted in **bold green**.

Method	Loss	Cityscapes	SMIYC RA-21		SMIYC RO-21		FS L&F		FS Static		Road Anomaly	
		mIoU ↑	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓	AuPRC ↑	FPR95 ↓
MSP	CE	72.20	29.10	62.51	2.71	64.97	1.75	50.76	7.47	41.82	12.43	82.49
	FL	72.20	27.94	66.24	2.94	64.90	**2.10**	47.66	7.70	41.84	12.23	82.50
	LN + CE	71.70	27.33	63.32	2.09	86.41	1.97	**46.74**	7.62	37.54	12.58	80.56
	IMP + CE	27.73	**37.06**	**52.95**	**3.81**	**26.68**	0.91	58.68	**12.21**	36.75	**19.66**	**65.20**
	LN + FL	71.74	27.49	61.59	2.09	83.61	1.84	47.77	7.95	**36.25**	12.37	79.61
	IMP + FL	17.79	33.55	63.28	3.52	45.70	0.87	51.02	4.18	63.43	11.99	78.85
MaxLogit	CE	72.20	38.32	59.34	4.63	48.44	3.30	45.49	9.50	40.30	15.58	73.25
	FL	72.20	39.13	60.50	6.37	33.73	3.54	47.32	8.89	36.77	18.19	70.89
	LN + CE	71.70	34.10	57.83	3.26	79.69	**3.86**	**40.27**	9.63	34.68	14.75	73.84
	IMP + CE	27.73	**41.05**	**50.93**	**6.56**	**25.49**	0.80	74.15	**14.98**	34.03	**19.35**	**63.76**
	LN + FL	71.74	34.17	56.10	3.29	76.80	3.51	41.07	10.15	**33.26**	14.61	73.05
	IMP + FL	17.79	35.03	62.71	5.24	52.61	0.77	41.73	4.30	61.55	12.13	79.81
Max Entropy	CE	72.20	31.01	62.59	3.05	65.60	2.58	50.37	8.83	41.52	12.68	82.63
	FL	72.20	29.58	66.58	3.23	65.62	**3.41**	47.23	9.33	41.45	12.57	82.77
	LN + CE	71.70	28.31	63.20	2.24	87.35	3.01	**46.40**	9.03	37.17	12.65	80.79
	IMP + CE	27.73	**34.47**	**49.91**	**20.40**	**15.40**	1.00	60.02	**11.73**	52.73	**15.65**	**75.13**
	LN + FL	71.74	28.55	61.50	2.23	84.60	2.75	47.41	9.42	**35.85**	12.42	79.87
	IMP + FL	17.79	22.95	61.24	3.58	68.17	0.68	52.87	3.48	62.38	11.08	80.36
Mahalanobis	CE	72.20	30.88	74.49	**9.64**	52.43	**2.94**	55.23	**8.93**	**39.34**	13.53	79.63
	FL	72.20	32.23	**55.54**	5.40	**14.61**	0.79	67.70	4.74	55.86	**28.63**	**60.04**
	LN + CE	71.70	36.77	56.59	7.36	35.98	1.40	66.92	4.61	60.86	19.66	62.44
	IMP + CE	27.73	19.80	89.24	2.52	72.14	0.16	99.36	4.14	92.70	7.47	91.20
	LN + FL	71.74	**37.73**	56.04	7.29	33.11	1.31	67.38	4.66	61.41	19.45	60.28
	IMP + FL	17.79	30.28	71.60	3.85	94.34	0.73	**55.08**	2.00	77.14	10.82	89.65

Table 5. Average forward time comparison across networks, computed on the Cityscapes dataset using a T4 GPU

	ERFNet	**ENet**	**BiSeNet**
Forward time [ms]	25.98	42.89	18.16
FPS [Hz]	38.49	23.32	55.08

5. Conclusion

This paper explored the effectiveness of real-time anomaly segmentation methods in road scenes, focusing on evaluating pre-trained models (ENet, ERFNet, and BiSeNet) and enhancing them with various techniques, such as temperature scaling, void classification, and advanced loss functions.

MaxLogit emerged as the most robust method for anomaly detection, consistently outperforming alternatives due to its ability to effectively distinguish between similar classes. Additionally, temperature scaling proved to be a valuable enhancement, improving the detection performance with optimal results achieved at a temperature value of 1.85. This indicates the importance of calibration in mitigating overconfidence in model predictions.

Incorporating the void class into fine-tuned models enabled explicit anomaly modeling, with BiSeNet demonstrating superior precision-recall metrics and ERFNet excelling at minimizing false positives. These results underline the benefits of leveraging existing dataset structures to enhance anomaly detection capabilities. Moreover, experimenting with advanced loss functions revealed that the Enhanced Isotropy Maximization Loss (IsoMax+) combined with Cross-Entropy Loss improved out-of-distribution detection performance, though its effectiveness varied across datasets, emphasizing the need for task-specific optimization.

From a computational perspective, BiSeNet exhibited the highest efficiency, achieving the fastest processing speeds while maintaining competitive segmentation accuracy. This makes it particularly suitable for real-time applications such as autonomous driving.

This study provides valuable insights into designing robust and efficient anomaly segmentation systems for real-world scenarios. Future research could focus on refining model reliability through advanced calibration techniques and metrics, expanding training datasets to enhance robustness in diverse scenarios, and adopting pruning and quantization methods to reduce model size and latency. Additionally, leveraging self-supervised pre-trained models and other foundational backbones could provide stronger generalization priors, improving performance across a wider range of environments.

References

[1] Hermann Blum, Paul-Edouard Sarlin, Juan I. Nieto, Roland Siegwart, and Cesar Cadena. The fishyscapes benchmark: Measuring blind spots in semantic segmentation. *CoRR*, abs/1904.03215, 2019. 5

[2] Robin Chan, Krzysztof Lis, Svenja Uhlemeyer, Hermann Blum, Sina Honari, Roland Siegwart, Mathieu Salzmann, Pascal Fua, and Matthias Rottmann. Segmentmeifyoucan: A benchmark for anomaly segmentation. *CoRR*, abs/2104.14812, 2021. 5, 6

[3] Robin Chan, Matthias Rottmann, and Hanno Gottschalk. Entropy maximization and meta classification for out-of-distribution detection in semantic segmentation. *CoRR*, abs/2012.06575, 2020. 3

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. *CoRR*, abs/1604.01685, 2016. 1, 4, 5

[5] Terrance DeVries and Graham W. Taylor. Learning confidence for out-of-distribution detection in neural networks, 2018. 1

[6] Tsung-Yi Lin e Priya Goyal e Ross B. Girshick e Kaiming Lui e Piotr Dollár. Perdita focale per il rilevamento di oggetti densi. *CoRR*, abs/1708.02002, 2017. 4

[7] Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Joe Kwon, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings, 2022. 3

[8] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *CoRR*, abs/1610.02136, 2016. 2

[9] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *ArXiv*, abs/1807.03888, 2018. 3

[10] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks, 2020. 3

[11] Krzysztof Lis, Krishna K. Nakka, Pascal Fua, and Mathieu Salzmann. Detecting the unexpected via image resynthesis. *CoRR*, abs/1904.07595, 2019. 5

[12] David Macêdo and Teresa Bernarda Ludermir. Improving entropic out-of-distribution detection using isometric distances and the minimum distance score. *CoRR*, abs/2105.14399, 2021. 4

[13] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *CoRR*, abs/1606.02147, 2016. 1, 2

[14] Peter Pinggera, Sebastian Ramos, Stefan Gehrig, Uwe Franke, Carsten Rother, and Rudolf Mester. Lost and found: Detecting small road hazards for self-driving vehicles. *CoRR*, abs/1609.04653, 2016. 5

[15] Eduardo Romera, José M. Álvarez, Luis M. Bergasa, and Roberto Arroyo. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):263–272, 2018. 1, 2

[16] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining, 2016. 5

[17] Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. Mitigating neural network overconfidence with logit normalization, 2022. 4

[18] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. *CoRR*, abs/1808.00897, 2018. 1, 2

Appendix A. Visualization

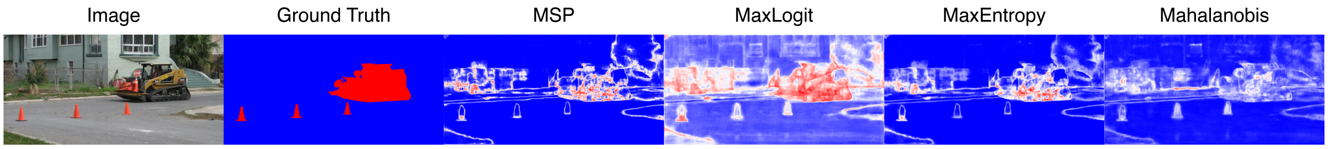

Figure 5. Visual comparison of *baseline* anomaly segmentation methods applied with ERFNet on the Road Anomaly dataset. The image on the left shows the input from the dataset, followed by the ground truth segmentation. The remaining columns display the outputs of MSP, MaxLogit, MaxEntropy, and Mahalanobis methods. The heatmap color scale ranges from **blue** (in-distribution) to **red** (anomaly).

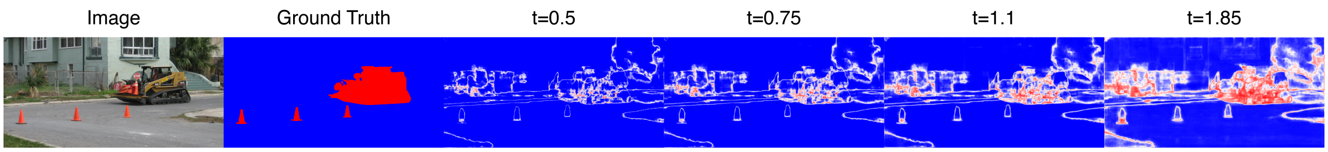

Figure 6. Visual comparison of anomaly segmentation using ERFNet and the MSP method with different *temperature scaling* values on the Road Anomaly dataset. The image on the left shows the input from the dataset, followed by the ground truth segmentation. The remaining columns display the outputs of MSP with different temperature values.

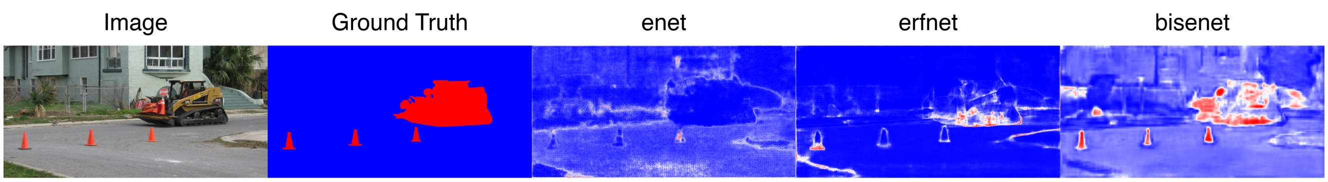

Figure 7. Visual comparison of anomaly segmentation with the three analyzed networks fine-tuned as *void classifiers*, applied on the Road Anomaly dataset. The image on the left shows the input from the dataset, followed by the ground truth segmentation. The remaining columns display the outputs of the networks fine-tuned as void classifiers.

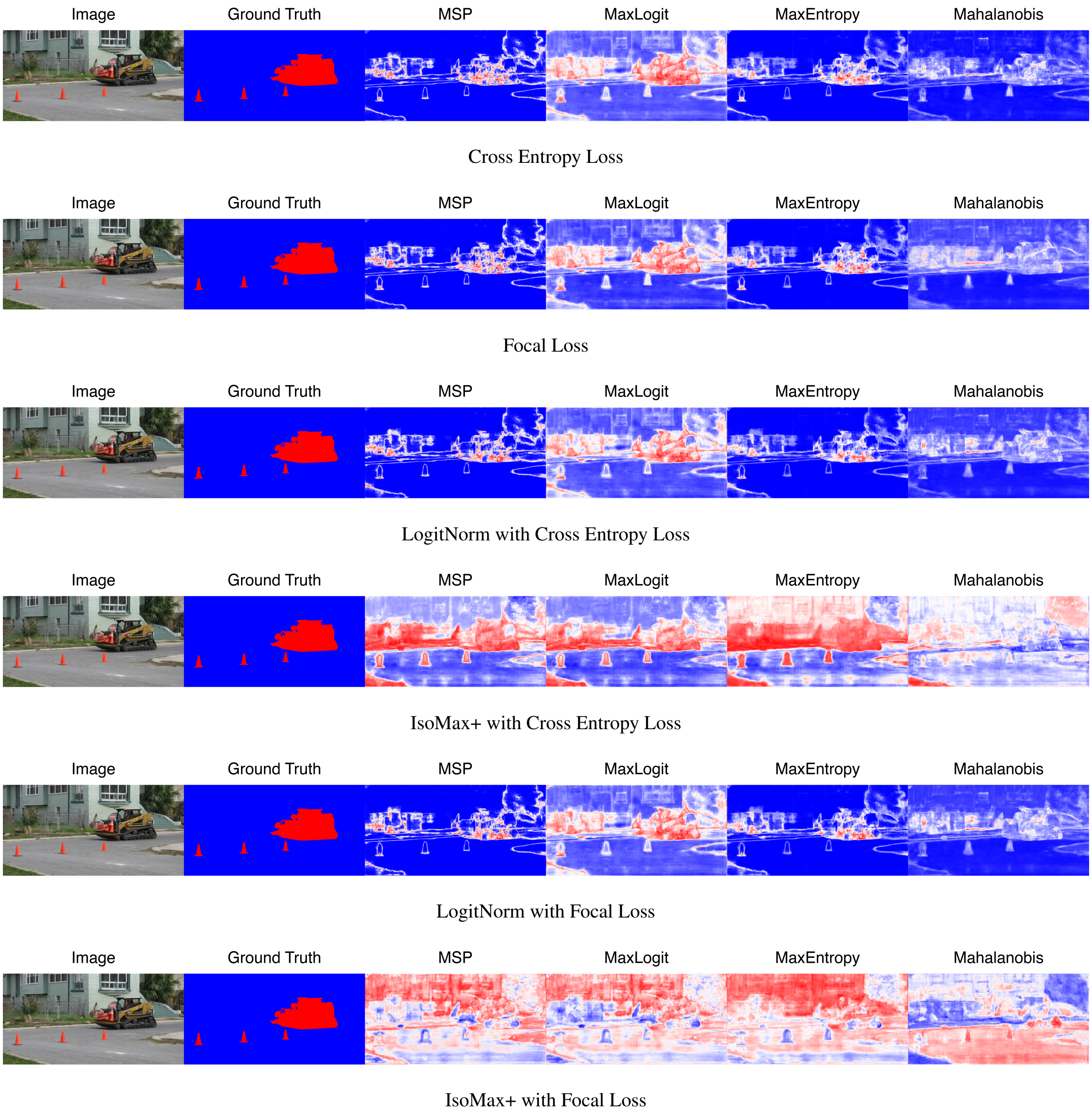

Figure 8. Visual comparison of anomaly segmentation methods and *loss functions* on the Road Anomaly dataset. The image on the left shows the input from the dataset, followed by the ground truth segmentation. The remaining columns display the outputs of MSP, MaxLogit, MaxEntropy, and Mahalanobis methods for each row. Rows correspond to different loss functions used during training: CrossEntropy, Focal Loss, LogitNorm with CrossEntropy, IsoMax+ with CrossEntropy, LogitNorm with Focal Loss, and IsoMax+ with Focal Loss.