A Comparative Study for Open Set Semantic Segmentation Methods

Anderson Brilhador, Matheus Gutoski, André Eugênio Lazzaretti, Heitor Silvério Lopes
Bioinformatics and Computational Intelligence Laboratory – LABIC/CPGEI
Federal University of Technology – Paraná (UTFPR)
Emails: {andersonbrilhador, matheusgutoski, andrelazzaretti, hslopes}@gmail.com
{brilhador, lazzaretti, hslopes}@utfpr.edu.br

***Abstract*—Typical semantic segmentation methods do not recognize unknown pixels during the test or deployment stage. This capability is critical for open-world environment applications where unseen objects appear all the time. Recently, to solve those limitations, Open Set Semantic Segmentation (OSSS) was introduced. This task aims to produce known and unknown pixels semantic segments. However, due to its recent introduction, few works are found in the literature, and consequently, few datasets are publicly available. This work carried out a comparative study between the existing OSSS methods on a new synthetic dataset of images and the well-known PASCAL VOC 2012 dataset. The compared methods include SoftMax-T, OpenMax-based, and OpenIPCS. The results are encouraging and show some of the advantages and main limitations of each technique. However, in general, they demonstrate that the problem of OSSS remains open and demands further research aiming at real applications, such as autonomous driving and robotics.**

***Index Terms*—Open Set Recognition, Open World Problem, Semantic Segmentation, Synthetic Geometry Dataset**

I. Introduction

In recent years, multi-class semantic segmentation has been boosted by advances in deep learning models [1]–[3]. This task aims to divide an image into multiple semantically meaningfull regions. This is done by classifying each image pixel in one of the predefined classes at the training step. There are several real-world applications of semantic segmentation, for instance, autonomous driving and robotics, since they are based on scene understanding to perform actions [4]–[7].

However, most existing approaches focus on a fixed number of known classes. In other words, those models are trained and evaluated on closed set conditions, which means that all test samples are present during training. This assumption can be easily violated in open-set problems, in which new objects might emerge unexpectedly after training. This scenario is particularly evident in autonomous driving, for instance, considering that it is practically impossible to collect and annotate a priori all instances found in the real-world. The ideal setting is to segment unknown regions of images in the test step, besides the known classes, informing users which image regions do not belong to the labels already learn during the training of the model. This task can be called Open Set Semantic Segmentation (OSSS).

Due to the recent introduction of OSSS, few works in the literature approach this problem. The OpenPixel proposed in [8] consists of patchwise training of a Convolutional Neural Network (CNN) and the definition of a probability threshold after the SoftMax layer. Pixels with class probability less than the threshold are labeled as unknown. In [9], the authors proposed a non-parametric statistic test to identify the presence of unknown regions in the prediction confidence map, obtained from the SoftMax layer of a trained CNN over the closed set. After this identification, an adaptive threshold is applied to the prediction confidence histogram to separate known and unknown regions in the image. In [10], [11], an OpenMax-based approach [12] is presented. It consists of replacing the last activation layer of a semantic segmentation network, usually the SoftMax layer, with the OpenMax layer, estimating the input probability, and labeling it as known or unknown.

Similarly, in [10], the authors presented a method based on principal components of the CNN's internal features to segment unknown regions, named Open Principal Component Scoring with Incremental Training (OpenIPCS). Those methods were applied to a set of aerial images, limiting the performance analysis in other applications. As a matter of fact, a detailed comparison pointing out the limitations of each method is still lacking in the literature.

Due to the importance of those methods for real-world applications, this work proposes a comparative study of the following state-of-the-art OSSS methods: SoftMax-Thresholding (SoftMax-T), OpenMax-based, and OpenIPCS in two distinct sets of images. The first is a set of synthetic images, and the other includes images with high intra-class and inter-class variability indexes. Hence, the main contributions of this work are:

- Building a new synthetic image dataset for OSSS task;
- Establishing two new benchmarks for training and evaluation protocol in OSSS tasks;
- Investigating the limitations of the state-of-the-art OSSS methods;
- Reassessing the main existing challenges for the OSSS task.

This work is organized as follows. Section II presents the OSSS problem and the methods used in this work. Section III shows the datasets used to compare the OSSS approaches. Section IV describes the configurations to conduct the experiments. Section V presents the quantitative and qualitative results obtained from the experiments. Section VI presents the

main challenges observed in the OSSS and, finally, Section VII presents the conclusions and future research directions.

II. Background

In this Section, we present the OSSS task and an overview of the main methods currently used in the literature: A) SoftMax-T; B) OpenMax-based; and C) OpenIPCS.

A. Open Set Semantic Segmentation

OSSS is a derivation of the Open Set Recognition problem introduced in [13]. The authors proposed the concept of openness, assuming that even knowing all classes during the supervised training, one should still consider the possibility of unknown samples appear later, during the prediction step. In summary, practical recognition systems must consider three basic categories, which are shown in Table I.

TABLE I
THE THREE BASIC CATEGORIES OF OSSS.

Category	Description
Known known classes (KKCs)	Classes with distinctly labeled positive training samples (also serving as negative samples for other KKCs).
Known unknown classes (KUCs)	Labeled negative samples, not necessarily grouped into meaningful categories, such as background [14] and universum [15] * classes.
Unknown unknown classes (UUCs)	Classes without any available information during training.

* The universum [15] classes represent the samples that do not belong to either class of interest for the specific learning problem.

The objective of OSSS is to classify each pixel of an image into one of the existing known classes (KKC and KUC) or the unknown class (UUC). The UUC can often combine many other classes that do not appear in the training dataset. Therefore, the segmentation model must be trained in a database without the presence of unknown classes [9].

B. SoftMax-T

Most DNNs use the SoftMax layer to classify input data. The SoftMax layer assigns probabilities to each class in a problem of several known classes. The class with the highest probability is assumed to be the predicted class.

A variant of SoftMax, named SoftMax-T, applies a threshold to the probability obtained by the SoftMax layer, following the premise that lower probabilities can be motivated by UUCs [10]. Thus, SoftMax-T consists of defining a Minimum Confidence Threshold (MCT) to classify the input data into one possible known class. Input data with a probability above the MCT are classified as unknown classes.

C. OpenMax-based

OpenMax was initially proposed for the Open Set Recognition task [12]. However, recently, OpenMax was applied to the OSSS task [10], [11]. This method consists of replacing the SoftMax layer with a layer called OpenMax that incorporates the explicit recognition of unknown classes using the extreme

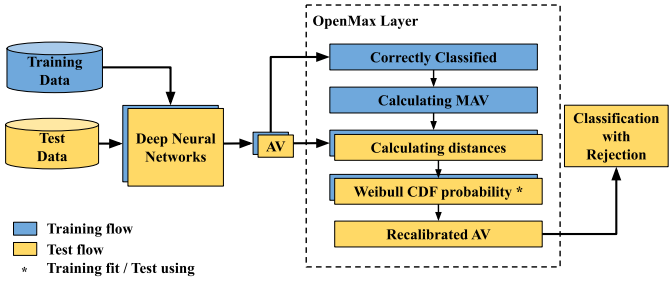


Fig. 1. Flow for adjusting and using the OpenMax-based method.

value theory [16]. Figure 1 shows a schematic diagram of this approach.

Based on a visual inspection, in [12], the authors found that the output of the last hidden layer before the SoftMax layer, called the activation vector (AV), provides considerable discriminative information between known and unknown classes. The dimensionality of the AV is equal to the number of known classes. Thus, the authors proposed that each known class can be represented as Mean Activation Vector (MAV), with the mean being calculated only on correctly classified training samples. Subsequently, the Euclidean distance of each correctly classified AV, for each obtained MAV, is calculated. The largest distance values of each MAV (tailsize) are used to fit the Weibull Cumulative Distribution Function ($\text{CDF}_{Weibull}$) for each known class.

In the test step, the distance of the AV to the MAVs of the known classes is calculated. Those distances are used to calculate the $\text{CDF}_{Weibull}$ probability of each sample belong a determined known class. Later, together with weighting weights (alpha), they are used to recalibrate the AV, by incorporating the probabilities of the unknown class. In this method, a MCT on the probabilities obtained by the OpenMAX layer is also used, working similarly to the SoftMax-T method.

D. OpenIPCS

The OpenIPCS [10] was inspired by the Conditional Gaussian Distribution Learning (CGDL) proposed by [17].It is a Variational Autoencoder (VAE) model for conditional Gaussian distribution estimation, capable of learning conditional distributions of known classes and rejecting unknown examples. It uses more than one activation layer, replacing the VAE with the Principal Component Analysis (PCA) and performing the adjustment of the generative model with validation data instead of training samples.

Shwarts and Tishby [18] demonstrated that the deeper a layer is placed in a deep neural network, the closer to the label space the activation features are. This approach is different from the OpenMax-based, which considers only the last activation layer to adjust the $\text{CDF}_{Weibull}$ for each KKC. In addition to the last layer, OpenIPCS takes into account the enabling characteristics of the previous layers, combining low-level and high-level semantic information.

Then, for each output pixel, a corresponding activation vector is constructed by concatenating the activation layers of the network. Such a concatenation produces high-dimensional and redundant feature vectors due to the hundreds or thousands of activation channels present in the CNN and the fully convolutional network (FCN) layers [2], [19].

As described by Tipping and Bishop [20], PCA, in addition to dimensionality reduction, can also be used as a probability density estimator with Gaussian priors. These features allow OpenIPCs to use PCA as a generative model (G) for novelty detection while solving the high-dimensionality problem of feature vectors.

This G is incrementally adjusted using only the validation images. This process consists of adjusting the PCA with a batch of samples belonging to the validation set. The classification step using the G consists of projecting the main components of the test images in the latent space obtained by adjusting the PCA on the validation images and, then, performing the inverse process. At the end, the difference between the original feature vector and the one after the projection is calculated. Consequently, pixels of known classes have low difference values, while pixels belonging to the unknown class have high difference values. The overall process is illustrated in Fig. 2.

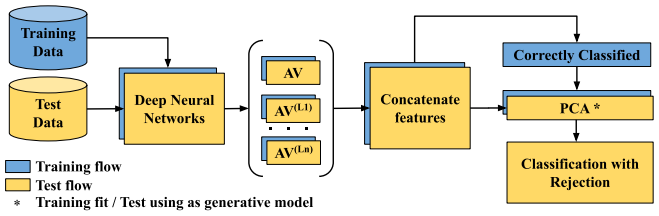


Fig. 2. Flow for adjusting and using the OpenIPCS.

III. Datasets

Since OSSS has been underexplored in the literature, there is no publicly available reference dataset for this task. In this Section, we present the datasets used in this work.

A. Synthetic Geometry Dataset

The Synthetic Geometry Dataset (SGDataset) is a synthetic dataset, inspired by [21], in which images are procedurally generated by drawing geometric shapes randomly. The dataset consists of three geometric shapes (circle, rectangle, and triangle) and a completely white background class.

Due to its procedural characteristic, the dataset is easily configurable for closed and open set segmentation tasks, with few limits for image generation. To create a closed set dataset, define the number of images that will be generated and the minimum and the maximum number of objects that will be drawn per image. To create an open set dataset define one or more geometric shapes as a UUC and further define if the images are for training or testing. This is important as unknown objects must not be present in the training stage. To ensure the reproducibility of the experiments, it is enough to define the same random seed. The process of creating an image of the dataset is summarized in the Algorithm 1.

Algorithm 1: Generation of a SGDataset image.

Input: Requires the minimum (min) and the maximum (max) number of geometric objects in the output image.
Input: Requires the number of known classes (k).
Input: Requires a new input image (x) and a ground truth (y) – are filled with a values 255 and 0, respectively.
Output: Returns the generated image (x) and ground truth (y).

$n = random(min, max)$
for $i = 0...n$ **do**
  $c = random(0, k + 1)$
  if $c < k$ **then**
    $x, y = draw\_known\_class(x, y, c)$
  else
    $x, y = draw\_unknown\_class(x, y, c)$
  end if
end for
return x, y

Geometric shapes are scaled based on a percentage of the input width image (see Table II).

TABLE II
Percentage ranges of input image width for defining the dimensions of the geometric shapes.

Class	Percentage range
Rectangle	width: [7.5%, 15%] , height: [15%, 30%]
Circle	radius: [7.5%, 15%]
Triangle	radius: [7.5%, 15%]

Object colors can vary randomly in a color space defined by the Matplotlib library, as shown in Figure 3.

Fig. 3. Color Map for the SGDataset.

B. PASCAL Visual Object Classes 2012 Dataset

The PASCAL Visual Object Classes (VOC) 2012 dataset [22] consists of 10,583 training images (including the extra annotations provided by [23]) and 1,449 validation images, totaling 12,031 images with pixel-level annotations. The dataset contains 20 well-known classes and one background class. The known classes were mapped to categories according to Table III to increase intra-class and inter-class variability, in addition to simplifying the experiments performed in this work.

IV. Experiment Setup

Experiments were performed on a workstation with an Intel Core-i5 9600K processor, 64GBytes of RAM, and a Nvidia RTX 3090 GPU, running Ubuntu 18.04 LTS operating system. The framework developed in this work was built using Python. The neural network architecture used was proposed by [10], a FCN with DenseNet-121 [2] pre-trained on ImageNet [24]

TABLE III
MAPPING OF PASCAL VOC 2012 CLASSES.

Original classes	Mapped class
Background, Bottle, Chair, Dining table, Potted plant, Sofa, TV/monitor	Background
Person	Person
Bird, Cat, Cow, Dog, Horse, Sheep	Animal
Aeroplane, Bicycle, Boat, Bus, Car, Motorbike, Train	Vehicle

was built in PyTorch. The implementation of PCA was build with the scikit-learn.

A. Overview

Figure 4 presents a block diagram of the proposed method used to carry out the experiments. The process is detailed as follows. First, the open set dataset is split into two: train and test data (see Section IV-B). The next step is to perform the closed set flow for obtaining the best model in the closed set (see Section IV-D). In the open set flow, the OSSS methods are trained to identify unknown samples (see Section IV-E). The last step is to compute the evaluation metrics (see Section IV-F).

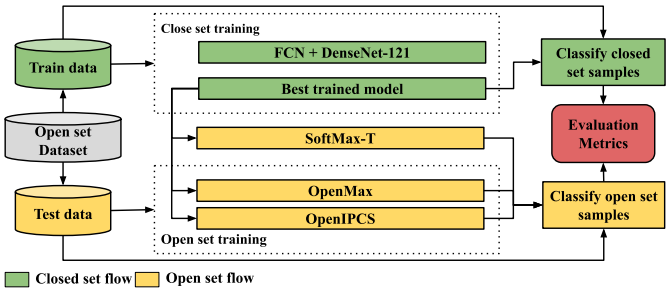


Fig. 4. Block diagram of the proposed method.

B. Open Set Dataset

The open set dataset is split into train and test data to be consistent with the OSSS task in which data from the UUC need not be present in the training step. Hence, PASCAL VOC 2012 images with UUC pixels were removed from the train data and allocated to the test data. A percentage of 20% of the images without UUC pixels were also allocated to compose the test data. Table IV shows the number of images by classes for training and testing.

In the SGDataset, the division was automatically accomplished at the creation time, as described in Section III-A. A total of 2000 and 500 images were used for training and testing data, respectively.

C. Pre-processing

In the first stage of pre-processing, the images were resized to 224×224 pixels due to the input size requirement of the convolutional model used in this work.The second step was to apply data augmentation techniques to improve segmentation results [25]. The last pre-processing step was to normalize the

TABLE IV
AMOUNT OF IMAGES USED IN THE EXPERIMENTS WITH THE PASCAL VOC 2012 DATASET.

UUC	Train data	Test data
Person	6149	5882
Animal	6136	5895
Vehicle	6312	5719

images using the mean (μ) and standard deviation (σ) of the images available in ImageNet [24], as shown in the Eq. 1:

$$z_i = \frac{x_i - \mu_i * 255}{\sigma_i * 255}, \tag{1}$$

in which x_i is the RGB channels of the input image, z_i is the channels normalized by $\mu = 0.485, 0.456, 0.406$ and $\sigma = 0.229, 0.224, 0.225$. Pre-processing was applied to all images used in this work.

D. Closed-set training

The first step is to split train data using the cross-validation with $k = 3$. The low value of k was chosen to increase the proportion of samples from known and unknown classes in the testing folds.

The second step was to apply a data augmentation method to the training folds before applying the operations defined in Section IV-C. We used the Albumentations[1] library to perform the transformations shown in Table V, applied with a probability of 20%.

TABLE V
DATA AUGMENTATION TRANSFORMATIONS.

Transformations	Parameters
Shift, Scale e Rotation	The shift and scaling factor range: $[-0.5, 0.5]$. Rotation factor range: $[-90, 90]$.
Random Brightness and Constrast	The factor range for brightness and contrast: $[-0.3, 0.3]$.
GridDropout [26]	The pixels of the removed input image region are filled with a value of 255.

Finally, the model was trained with a batch size of 16 images until the cross-entropy loss stagnates for 5 consecutive epochs. For this process, we used the Adam optimizer with an initial learning rate of 0.0001, and weight decay coefficient of 0.00001. The training process was repeated until all folds are used in the testing.

E. Open-set training

The OSSS methods studied in this work depend on a semantic segmentation model for open set recognition. Thus, the model with the highest (m)IoU (mean Intersection over Union) in the closed set was selected for this step. Due to the simplicity of the SoftMax-T method in thresholding the output of the SoftMax function, it is not necessary to carry out any training process for this method. OpenMax-based and OpenIPCS need to adjust the $\text{CDF}_{Weibull}$ and G, respectively,

[1] https://albumentations.ai/

for classification of known and unknown pixels, as described in Section II. For the evaluation of the methods, 20% of the images were reserved from the test data presented in Table IV.

F. Evaluation

In this work, the following metrics were used for evaluating the proposed approach: Precision, Recall and IoU, defined by Equations 2, 3 and 4, respectively. These metrics are commonly used to evaluate closed set segmentation methods and have been recently explored in the context of open set segmentation [9]–[11].

$$Precision_i = \frac{TP_i}{TP_i + FP_i}, \tag{2}$$

$$Recall_i = \frac{TP_i}{TP_i + FN_i}, \tag{3}$$

$$IoU_i = \frac{TP_i}{TP_i + FP_i + FN_i}. \tag{4}$$

where TP_i, FP_i, FN_i are true positives, false positives, and false negatives for each class i.

Precision is a measure that indicates the proportion of correctly segmented pixels. Recall is a measure of completeness that specifies the proportion of positive pixels in the ground truth that is also identified as positive by the segmentation. The IoU is calculated for each semantic class. This measure seeks to assess the similarity between two finite sets, in this case, ground truth and the segmentation predicted by the model. The mIoU is the average of all classes.

V. Experimental Results and Discussion

This Section is divided into two parts: quantitative and qualitative results, and both parts include the results obtained by each open set segmentation method: SoftMax-T, OpenMax-based, and OpenIPCS.

A. Quantitative Results

Tables VI and VII show the results of the three OSSS methods evaluated in this work using the SG-Dataset and PASCAL VOC 2012 image datasets, respectively. The parameters presented were defined through an exhaustive search in the following values: threshold: $[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]$, alpha: $[2, 3]$ and tail-size: $[20, 40, 60, 80, 100]$.

The average results obtained by the methods in the SG-Dataset, which has a more controlled set of images, suggest that OpenIPC presented superior results in terms of accuracy, recall, and IoU. This fact indicates that it is the most suitable method among those proposed for this application. However, analyzing the results obtained on the PASCAL VOC 2012 dataset, which has a more complex set of images with high intra-class and inter-class information variability, the superiority presented in the previous experiment was not repeated. In general, the results found are similar between the three models and far from being considered satisfactory for practical OSSS tasks.

Considering only the unknown classes, in general, OpenIPCs was superior to SoftMax-F and OpenMax. For the SGDataset, in particular, the UUC: circle presented similar results in terms of precision between the three methods. However, recall and IoU were 80% and 58.62%, respectively, with the OpenIPCS method, indicating that the predictions performed by the method were more consistent with respect to pixels of unknown class. That is, the pixels predicted to be unknown by the segmentation are actually unknown pixels in the ground truth. Similar behavior happens in PASCAL VOC 2012, with UUC: vehicle, with lower recall and IoU values, 40% and 30.04%, respectively.

Analyzing only the results obtained on the PASCAL VOC 2012, we can highlight the relationship between the animal and person classes. When one of the classes was used as UUC, the other obtained low values for the metrics. This indicates a possible confusion between the OSSS methods, probably due to the similarity between the classes.

The best result obtained among all the analyzed experiments was in the UUC: triangle class. The three methods achieved results above 72% in all metrics analyzed. The best method was OpenIPCs which averaged 87.72%, 92.09%, and 85.47% for accuracy, recall, and IoU metrics, respectively. The semantic segmentation network learns different elements of the object to make predictions such as shape, texture, color, etc. The UUC: triangle class stands out for the coloring of the objects defined in the section III, which is slightly different from the other objects. In the next section, we will explore the visual characteristics of images further.

B. Qualitative results

Tables VIII and IX present some samples of the best segmentations generated by the methods discussed in this work. In the Figures it is possible to observe that the quantitative assessment is reflected in the qualitative results. In general, all methods have limitations to precisely segment objects of the UUC classes.

OpenIPCS stands out compared to SoftMax-T and OpenMax-based, specially in situations where there is not a large variety of intra-class pixels. That is, in all UUCs of the SGDataset and the UUC: vehicle of the PASCAL VOC 2012 dataset. On the other hand, OpenIPCS produces artifacts of the UUC class, even in the best results obtained by the method. This feature can also be seen in his original work [10].

The results generated by SoftMax-T and OpenMax-based are similar to each other and inferior to OpenIPCS in more controlled situations, as in the case of the SGDataset. Another important factor is the difficulty of these methods in dealing with the boundaries between classes in images. In general, predicting the pixels in this region as belonging to the UUC. This result is due to the characteristic of the methods in manipulating the SoftMax layer outputs to include the UUC prediction. The SoftMax layer produces class predictions with lower certainty for pixels belonging to the boundary regions between objects [10].

TABLE VI

PERFORMANCE OF CLOSED SET SEMANTIC SEGMENTATION AND OPEN SET SEMANTIC SEGMENTATION ACHIEVED ON THE GEOMETRY DATASET. BOLD VALUES INDICATE THE BEST RESULTS OF (M)IOU.

UUC: Circle

KKCs	Closed set	SoftMax-T			OpenMax-based			OpenIPCS		
	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU
Background	91.37%	99.64%	98.54%	94.57%	99.75%	98.31%	**95.38%**	99.98%	83.91%	83.63%
Rectangle	94.48%	39.43%	89.42%	48.20%	39.64%	88.92%	48.24%	72.41%	91.43%	**76.00%**
Triangle	92.64%	98.11%	83.36%	82.96%	98.44%	82.30%	81.99%	96.44%	87.22%	**86.94%**
Unknown	-	32.52%	12.17%	9.63%	33.41%	14.22%	11.07%	33.96%	80.00%	**58.62%**
Average	92.83%	67.42%	70.87%	58.84%	67.81%	70.94%	59.17%	75.70%	85.64%	**76.30%**
Parameters		Threshold: 0.8			Tailsize: 80, Alpha: 3, Threshold: 0.8			Threshold: 0.8		

UUC: Rectangle

KKCs	Closed set	SoftMax-T			OpenMax-based			OpenIPCS		
	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU
Background	93.04%	99.77%	97.11%	93.96%	99.84%	96.59%	**94.47%**	99.94%	88.34%	87.51%
Circle	96.23%	58.47%	93.40%	49.58%	58.92%	92.78%	49.66%	79.20%	94.79%	**72.75%**
Triangle	92.58%	96.51%	81.84%	81.29%	97.46%	78.68%	78.32%	90.99%	87.63%	**87.06%**
Unknown	-	16.17%	10.09%	8.06%	16.05%	12.26%	9.41%	29.77%	70.00%	**54.98%**
Average	93.95%	67.73%	70.61%	58.22%	68.07%	70.07%	57.97%	74.98%	85.19%	**75.57%**
Parameters		Threshold: 0.8			Tailsize: 100, Alpha: 3, Threshold: 0.8			Threshold: 0.7		

UUC: Triangle

KKCs	Closed set	SoftMax-T			OpenMax-based			OpenIPCS		
	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU
Background	92.52%	99.22%	97.27%	84.44%	99.45%	96.41%	**87.17%**	99.05%	97.59%	83.35%
Rectangle	92.14%	87.79%	76.98%	67.31%	88.89%	72.25%	64.36%	90.98%	93.42%	**88.40%**
Circle	96.02%	99.40%	88.83%	88.28%	99.72%	86.05%	85.86%	95.74%	97.35%	**94.75%**
Unknown	-	39.58%	70.76%	52.03%	35.85%	77.45%	53.51%	65.10%	80.00%	**75.38%**
Average	93.56%	81.50%	83.46%	73.02%	80.98%	83.04%	72.72%	87.72%	92.09%	**85.47%**
Parameters		Threshold: 0.9			Tailsize: 100, Alpha: 3, Threshold: 0.9			Threshold: 0.8		

TABLE VII

PERFORMANCE OF CLOSED SET SEMANTIC SEGMENTATION AND OPEN SET SEMANTIC SEGMENTATION ACHIEVED ON THE PASCAL VOC 2012 DATASET. BOLD VALUES INDICATE THE BEST RESULTS OF (M)IOU.

UUC: Person

KKCs	Closed set	SoftMax-T			OpenMax-based			OpenIPCS		
	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU
Background	86.09%	89.16%	91.60%	57.64%	89.45%	91.36%	**58.10%**	88.87%	86.28%	53.15%
Animal	93.69%	70.14%	72.90%	66.72%	65.90%	77.99%	**69.71%**	59.03%	78.42%	68.41%
Vehicle	90.61%	88.13%	62.30%	60.87%	88.75%	61.64%	60.35%	75.35%	76.14%	**71.72%**
Unknown	-	41.66%	41.66%	25.88%	41.57%	41.04%	26.19%	39.05%	40.00%	**29.41%**
Average	90.13%	72.27%	67.12%	52.78%	71.42%	68.01%	53.59%	65.58%	70.21%	**55.67%**
Parameters		Threshold: 0.9			Tailsize: 20, Alpha: 3, Threshold: 0.8			Threshold: 0.4		

UUC: Animal

KKCs	Closed set	SoftMax-T			OpenMax-based			OpenIPCS		
	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU
Background	82.91%	93.72%	90.26%	67.47%	94.67%	89.17%	**69.14%**	93.81%	82.23%	60.78%
Person	87.77%	35.04%	68.42%	53.32%	37.26%	67.27%	53.78%	32.74%	76.60%	**57.91%**
Vehicle	87.80%	82.59%	69.56%	68.08%	85.56%	67.32%	66.24%	65.97%	72.89%	**69.13%**
Unknown	-	52.60%	49.78%	31.57%	53.11%	57.33%	**35.12%**	43.54%	50.00%	34.42%
Average	86.16%	65.99%	69.51%	55.11%	67.65%	70.27%	**56.07%**	59.02%	70.43%	55.56%
Parameters		Threshold: 0.8			Tailsize: 20, Alpha: 2, Threshold: 0.7			Threshold: 0.5		

UUC: Vehicle

KKCs	Closed set	SoftMax-T			OpenMax-based			OpenIPCS		
	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU	Precision	Recall	(m)IoU
Background	84.86%	81.58%	97.35%	47.45%	81.57%	97.46%	47.27%	86.58%	92.60%	**52.56%**
Person	88.59%	86.29%	70.31%	68.85%	85.20%	72.48%	70.88%	77.40%	77.23%	**73.92%**
Animal	91.08%	90.66%	83.17%	82.02%	89.97%	83.96%	**82.84%**	85.55%	74.41%	72.98%
Unknown	-	29.35%	6.87%	5.14%	32.09%	6.89%	5.32%	54.82%	40.00%	**30.04%**
Average	88.18%	71.97%	64.42%	50.87%	72.21%	65.20%	51.58%	76.09%	71.06%	**57.38%**
Parameters		Threshold: 0.8			Tailsize: 20, Alpha: 3, Threshold: 0.6			Threshold: 0.4		

TABLE VIII
SOME OF THE BEST VISUAL RESULTS SAMPLES OBTAINED ON THE SGDATASET. UUC IS REPRESENTED BY RED PIXELS.

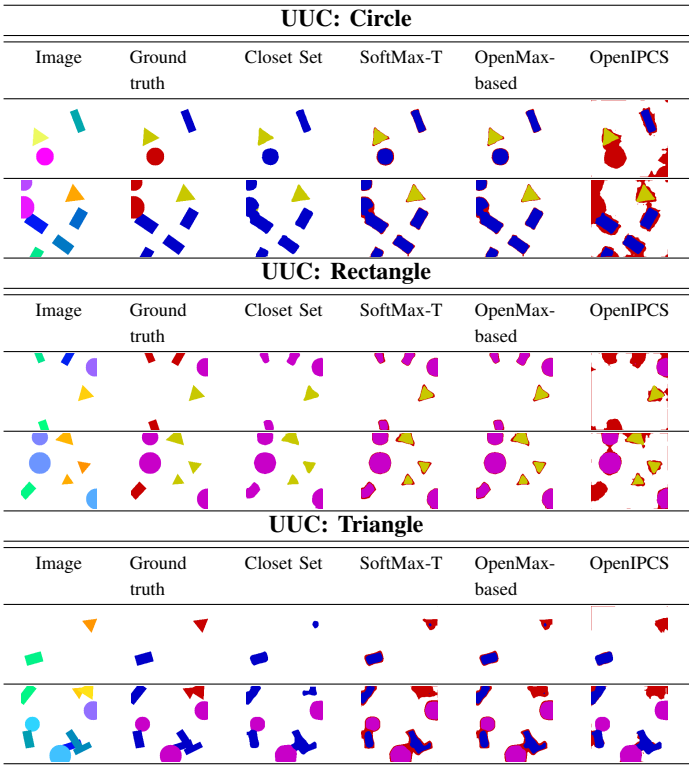
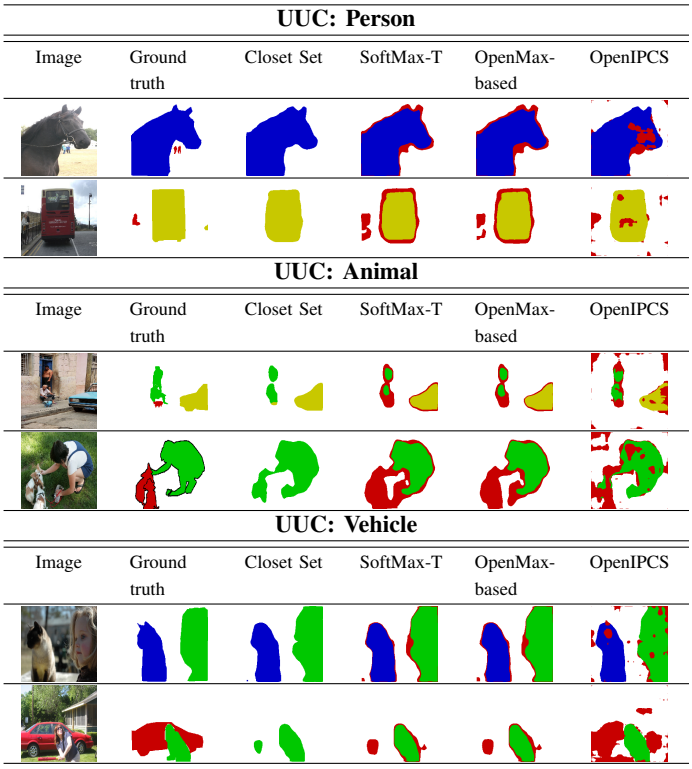


TABLE IX
SOME OF THE BEST VISUAL RESULT SAMPLES OBTAINED ON THE PASCAL VOC 2012. UUC IS REPRESENTED BY RED PIXELS.



VI. CHALLENGES IN OPEN SET SEMANTIC SEGMENTATION

Considering that the objective of OSSS is to precisely segment the pixels of the known and unknown classes, a detailed analysis of the results was carried out to demonstrate the main challenges of this task. In this review we expand and redefine the challenges summarized by [9]:

1) **Invisible objects in known classes that resemble the unknown class:** due to difficulties in annotating a large volume of images for semantic segmentation, several image regions end up being "ignored" in the labeling process. Typically, these regions were labeled as the background (known) class, making it reasonable for the OSSS model to classify objects of the background class as objects of the unknown class and vice-versa.
2) **Objects of unknown classes similar to known classes:** due to a large number of possible classes in the classification task, it is natural to find objects of known classes similar to some objects of unknown classes. In this challenge, we will expand the description proposed by [9], considering that the neural networks for segmentation use different aspects of the object (color, texture, shape, etc) to perform the task. A strong similarity in only one of these aspects can increase the difficulty for the model to classify pixels. For example, to the human eyes, we might think that the person class does not have animal-like appearance. However, the human hair is very similar to the animal hair, making it difficult for these parts to be classified correctly. This similarity can also be expanded to other objects such as sofas, rugs, etc.
3) **Precisely segmented objects of known and unknown classes:** this challenge incorporates several sub-problems that are found in closed-set segmentation and also occur in open-set segmentation. In closed-set segmentation, the models suffer from the sub-problem of overlapping between objects, making it difficult to accurately separate the boundaries between objects. Another common sub-problem is the small objects present in the images, which are generally classified as objects of other classes, such as the background class. All these sub-problems are inherited by the OSSS task. However, in open-set segmentation, we still have the challenge of accurately classifying regions of completely unknown objects, while dealing with the sub-problems derived from closed-set segmentation.

VII. CONCLUSION

In this work, we investigate the performance of the main OSSS methods applied to different image segmentation problems. Two main points were explored: (i) the quantitative and quantitative performance of the methods; and (ii) possible limitations of the methods.

Regarding the first point, the experiments showed that the methods found difficulties in dealing with the OSSS task. Only OpenIPCS presented promising results for particular

situations, e.g., where the unknown class had a high intra-class variation and a low inter-class variation.

In the second point, the main limitations of the SoftMax-T and OpenMax-based methods were linked to their formulation, which uses only one layer of the neural network to incorporate the recognition of the unknown. Possibly, this is causes a drawback in separating distinct, but similar, classes with pixels belonging to regions of boundaries between objects that have low predictive probability values, being commonly segmented as belonging to an unknown class. OpenIPCS mitigated the problem of pixels belonging to object boundaries by using more than one layer of the neural network to map the feature space of known classes. However, this method presents artifacts of the unknown class in the resulting image, probably due to the similarity of the pixels in specific points of the image with the unknown class.

In general, this work revealed that it is hard to implement the semantic segmentation methods in real-world applications, where the segmentation of the known and unknown classes must be done precisely. We also redefined the main challenges of OSSS, in order to collaborate with future work that can address these limitations. Finally, based on the experiments done, we consider that the OSSS task is still an open problem. Thus, it is expected that future works can improve the performance of the models and make this task viable for real-world applications. We believe that studies in Deep Metric Learning and Bayesian Deep Learning may boost the performance of the OSSS task.

ACKNOWLEDGMENTS

A. Brilhador thanks for support from the Federal Technological University of Paraná (UTFPR); M. Gutoski thanks Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) for the scholarship n° 141983/2018-3; H.S.Lopes thanks CNPq for the research grant n° 311785/2019-0 and Fundação Araucária for the financial support by means of PRONEX 042/2018.

REFERENCES

[1] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[2] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708.

[3] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. of the European Conference on Computer Vision*, 2018, pp. 801–818.

[4] M. Siam, M. Gamal, M. Abdel-Razek, S. Yogamani, M. Jagersand, and H. Zhang, "A comparative study of real-time semantic segmentation for autonomous driving," in *Proc.of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2018.

[5] A. Milioto, P. Lottes, and C. Stachniss, "Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in cnns," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 2229–2235.

[6] M. Hofmarcher, T. Unterthiner, J. Arjona-Medina, G. Klambauer, S. Hochreiter, and B. Nessler, "Visual scene understanding for autonomous driving using semantic segmentation," in *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer, 2019, pp. 285–296.

[7] M. H. Hamian, A. Beikmohammadi, A. Ahmadi, and B. Nasersharif, "Semantic segmentation of autonomous driving images by the combination of deep learning and classical segmentation," in *Proc. of the 26th International Computer Conference, Computer Society of Iran*, 2021, pp. 1–6.

[8] C. C. V. da Silva, K. Nogueira, and J. A. Oliveira, Hugo N Santos, "Towards open-set semantic segmentation of aerial images," in *Proc. of the IEEE Latin American GRSS & ISPRS Remote Sensing Conference (LAGIRS)*, 2020, pp. 16–21.

[9] Z. Cui, W. Longshi, and R. Wang, "Open set semantic segmentation with statistical test and adaptive threshold," in *Proc. of the IEEE International Conference on Multimedia and Expo (ICME)*, 2020, pp. 1–6.

[10] H. Oliveira, C. Silva, G. L. S. Machado, K. Nogueira, and J. A. Santos, "Fully convolutional open set segmentation," *ArXiv preprint*, vol. 2006.14673, 2020.

[11] Y. Liu, Y. Tang, L. Zhang, L. Liu, M. Song, K. Gong, Y. Peng, J. Hou, and T. Jiang, "Hyperspectral open set classification with unknown classes rejection towards deep networks," *International Journal of Remote Sensing*, vol. 41, no. 16, pp. 6355–6383, 2020.

[12] A. Bendale and T. E. Boult, "Towards open set deep networks," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1563–1572.

[13] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, "Toward open set recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 7, pp. 1757–1772, 2012.

[14] A. R. Dhamija, M. Günther, and T. E. Boult, "Reducing network agnostophobia," in *Proc. of the 32nd International Conference on Neural Information Processing Systems*, 2018, p. 9175–9186.

[15] J. Weston, R. Collobert, F. Sinz, L. Bottou, and V. Vapnik, "Inference with the universum," in *Proc. of the 23rd International Conference on Machine Learning*, 2006, pp. 1009–1016.

[16] W. J. Scheirer, A. Rocha, R. J. Micheals, and T. E. Boult, "Meta-recognition: The theory and practice of recognition score analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1689–1695, 2011.

[17] X. Sun, Z. Yang, C. Zhang, K.-V. Ling, and G. Peng, "Conditional gaussian distribution learning for open set recognition," in *Proc.of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 480–13 489.

[18] R. Shwartz-Ziv and N. Tishby, "Opening the black box of deep neural networks via information," *ArXiv preprint*, vol. 1703.00810, 2017.

[19] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Highway networks," *ArXiv preprint*, vol. 1505.00387, 2015.

[20] M. E. Tipping and C. M. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural Computation*, vol. 11, no. 2, pp. 443–482, 1999.

[21] Y. Liu, E. T. Psota, and L. C. Pérez, "Layered embeddings for amodal instance segmentation," in *Proc. of the International Conference on Image Analysis and Recognition*, 2019, pp. 102–111.

[22] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, 2015.

[23] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik, "Semantic contours from inverse detectors," in *Proc. of the IEEE International Conference on Computer Vision*, 2011, pp. 991–998.

[24] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[25] A. Brilhador, M. Gutoski, L. T. Hattori, A. de Souza Inácio, A. E. Lazzaretti, and H. S. Lopes, "Classification of weeds and crops at the pixel-level using convolutional neural networks and data augmentation," in *2019 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*. IEEE, 2019, pp. 1–6.

[26] P. Chen, S. Liu, H. Zhao, and J. Jia, "Gridmask data augmentation," *ArXiv preprint*, vol. 2001.04086, 2020.