The Visual Computer
https://doi.org/10.1007/s00371-023-02925-8

ORIGINAL ARTICLE

GOSS: towards generalized open-set semantic segmentation

Jie Hong[1,2] · Weihao Li[1] · Junlin Han[1,2] · Jiyang Zheng[4] · Pengfei Fang[5] · Mehrtash Harandi[3] · Lars Petersson[2]

Accepted: 26 May 2023

Abstract
In this paper, we extend Open-set Semantic Segmentation (OSS) into a new image segmentation task called Generalized Open-set Semantic Segmentation (GOSS). Previously, with well-known OSS, the intelligent agents only detect unknown regions without further processing, limiting their perception capacity of the environment. It stands to reason that further analysis of the detected unknown pixels would be beneficial for agents' decision-making. Therefore, we propose GOSS, which holistically unifies the abilities of two well-defined segmentation tasks, i.e. OSS and generic segmentation. Specifically, GOSS classifies pixels as belonging to known classes, and clusters (or groups) of pixels of unknown class are labelled as such. We propose a metric that balances the pixel classification and clustering aspects to evaluate this newly expanded task. Moreover, we build benchmark tests on existing datasets and propose neural architectures as baselines. Our experiments on multiple benchmarks demonstrate the effectiveness of our baselines. Code is made available at https://github.com/JHome1/GOSS_Segmentor.

Keywords Open-set semantic segmentation · Generic segmentation · Scene understanding

1 Introduction

Image segmentation has significantly progressed in the deep learning era, especially class-specific semantic segmentation (SS) [1–7]. The goal of the SS task is to predict the class label of each pixel in an image from a set of predefined object classes. Adjacent pixels naturally belong together to form a segment when they share the same object category. Despite the considerable improvement, most SS settings follow a closed-set assumption that training and test data come from the same set of *known* object classes [8–11]. However, this assumption is rarely the case in practice, and it limits the generalization of segmentation models to *unknown* classes which models do not see during training.

Open-set semantic segmentation (OSS) [12–17] has recently been proposed to relax the above assumption, which aims to segment an image containing both known and unknown object classes. Unlike SS, OSS identifies the unknown region where pixels belong to unknown classes.

Although OSS aims to identify pixels that do not belong to one of the known classes, it does not provide any further processing or analysis amongst those identified unknown pixels. We argue that such a setting of OSS may limit the broad usage of vision-based intelligent agents when they encounter unfamiliar scenes where several unknown object classes are adjacent to each other rather than separated. Consider a scenario where an intelligent agent enters a new scene, as shown in Fig. 1a. OSS leaves the whole unknown region as a large segment without further processing (see "black region" in

✉ Jie Hong
jie.hong@anu.edu.au

Weihao Li
weihao.li1@anu.edu.au

Junlin Han
u6835134@anu.edu.au

Jiyang Zheng
jiyang.zheng@anu.edu.au

Pengfei Fang
pengfei.fang@anu.edu.au

Mehrtash Harandi
mehrtash.harandi@monash.edu

Lars Petersson
lars.petersson@data61.csiro.au

1 Australian National University, Canberra, Acton 2601, Australia

2 Data61, CSIRO, Canberra, Acton 2601, Australia

3 Monash University, Melbourne, VIC 3800, Australia

4 University of Sydney, Sydney, NSW 2006, Australia

5 Southeast University, Nanjing 211189, China

Published online: 23 June 2023

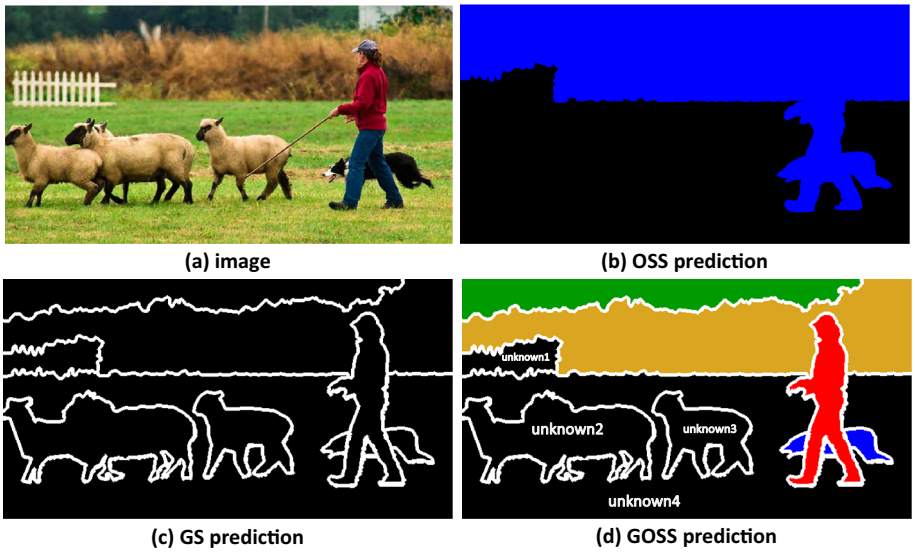


Fig. 1 Different tasks of image segmentation. For a given input image **a** that contains both known ("person", "dog" and "vegetation") and unknown objects ("sheep", "rail" and "grass"), we show: **b** open-set semantic segmentation (OSS) by pixel identification, **c** generic segmentation (GS) by pixel clustering, and **d** generalized open-set semantic segmentation (GOSS)

Fig. 1b). Insufficient information provided by the OSS setting might affect the decision-making of intelligent agents. Hence, we raise fundamental questions: how to improve OSS to generate richer representations for images in real-world scenes? Moreover, what is a more expressive and versatile image segmentation task beyond OSS? These questions are crucial in real-world applications like autonomous driving and robotics.

Towards the goal of better handling unknown regions in an image, inspired by the perception system of humans that they can jointly recognize the previously known objects and easily group unknown areas into different segments even though they do not know the categories of those unknown objects, this paper studies a new type of image segmentation task called *Generalized Open-set Semantic Segmentation (GOSS)*. It aims to classify pixels belonging to known classes and group the unknown pixels (see Fig. 1d for GOSS prediction). As we can see from Fig. 1d, "unknown rail" (or "unknown 1") and "unknown sheep" (or "unknown2") are segmented out from "unknown grass" (or "unknown4"). GOSS takes advantage of generic segmentation (GS), which groups pixels into segments sharing similarities [18–21, 21–23]. Compared to OSS, GOSS can detect more "objects" inside unknown regions. We specify two real-world applications where GOSS can help. First, again considering the example in Fig. 1a, self-driving cars may be assisted in avoiding potential obstacles if "unknown sheep" or "unknown rail" inside "unknown grass" can be found in the OSS prediction (see Figure b). Another possible practical example is that new detected "objects" inside unknown regions of images could help accelerate the data annotation process, especially when images from unfamiliar scenes are being labelled.

To enable intelligent agents to perform GOSS, we first build benchmarks using existing segmentation datasets, i.e. COCO-Stuff [11] and Cityscapes [9]. We split the full set of object categories into two sets: known classes and unknown classes. We keep the semantic annotations of known categories. For unknown categories, we use connectivity labelling [23] to convert their original semantic annotations into clustering ground truths. Along with the available datasets, a valid metric is also required to validate the quality of both OSS and GS. Although many existing metrics for segmentation tasks exist, such metrics are limited to measuring a single setting. In this work, we introduce a metric, termed GOSS Quality (GQ), which evaluates the segmentation quality of both known and unknown objects. Having the datasets and evaluation metrics at hand, we further establish a trainable framework, namely, GOSS SegmenTor (GST). The proposed GST adopts a dual-branch architecture with a shared backbone network. To perform the GOSS task, one branch conducts pixel classification, and the other performs pixel clustering. Moreover, to learn more discriminative embeddings and thus better process unknown objects, we leverage the pixel-wise contrastive learning loss into the training.

In summary, our contributions are as follows: (1) We present a new image segmentation task called GOSS, which jointly classifies known pixels and groups identified unknown pixels from OSS; (2) we propose the GQ metric that measures the quality of both pixel classification and pixel clustering, under open-set settings; (3) according to settings of GOSS, we build benchmarks by customizing existing datasets; (4) we show a simple yet effective baseline and its extended version, GST, to facilitate future research.

Table 1 Comparisons of different image segmentation tasks. Compared to traditional segmentation tasks, GOSS takes better care of unknown objects

Task	Known classes	Unknown classes
Generic segmentation	Cluster	Cluster
Open-set semantic segmentation	Classify	Identify
Generalized open-set semantic segmentation	Classify	Identify & cluster

2 Related work

Image segmentation is one of the most widely explored tasks in computer vision. Throughout image segmentation research, novel segmentation tasks have been crucial in driving research directions and innovations. We provide comparisons between our new setting and relevant older tasks in Table 1.

Open-set Semantic Segmentation (OSS). OSS, capable of identifying unknown objects, has developed significantly recently. Performing OSS is essential for intelligent agents as they work in open-set settings where many objects are never seen. A natural solution, studied in [12, 24, 25], determines the unknown regions by directly computing the anomaly score from logit or confidence vectors provided by the model classifier. Alternatively, synthesis approaches [13, 26–29] are proposed to detect unexpected anomalies from reconstructed images. In addition, the work [14] employs metric learning to learn more discriminative features and incrementally label novel classes using a human-in-the-loop approach. Beyond the existing OSS setting, the proposed GOSS performs holistic segmentation via classifying the known objects and clustering the unknown objects, providing more expressive information about the environment than OSS. With richer information, GOSS could benefit practical usage in real-world scenarios.

Generic Segmentation (GS). The task of GS is to find groups of pixels that "go together" [30]. In the early days of computer vision, the term "image segmentation" and the bottom-up general (non-semantic) segmentation share the same meaning. Recently, it is often called "generic segmentation" [21–23] to distinguish it from other segmentation tasks. The pipeline of early segmentation methods consists of first extracting local pixel features such as brightness, colour, or texture and then clustering these features based on, e.g. mean shift [19], normalized cuts [18], random walks [31], graph-based representations [20], or oriented watershed transform [21]. Learning-based image segmentation methods have now also become popular. DEL [32] learns a feature embedding corresponding to a similarity measure between two adjacent superpixels. Saacs et al. [22] propose pixel-wise representations that reflect how segments are related. Super-BPD [23] learns super boundary-to-pixel direction to provide a direction similarity between adjacent pixels. Comparing the performance of different image segmentation algorithms, public datasets such as BSDB [21, 33] provide human-labelled class-agnostic ground truth. However, they do not provide any semantic information.

Image Segmentation as a subtask. Image segmentation is often taken as a subtask jointly solved with other vision problems in a single framework [34–36], [37–42]. Panoptic segmentation [43–45] has recently become a standard image segmentation task by unifying semantic and instance segmentation.

3 Format and metric

3.1 Task format

Here, the task format for GOSS is formulated at a pixel level. For the ith pixel of an image, the GOSS output is defined as a pair $\mathbf{goss}_i = (s_i, g_i)$, where the classification label s_i indicates the pixel's semantic class and the clustering (or grouping) label g_i represents the cluster id. Suppose that there are N known semantic classes $\mathbf{L}^{kn} \in \mathbb{R}^N$ and an unknown class indicator $L^{uk} \in \mathbb{R}$, we have the semantic label set $\mathbf{L} = \{\mathbf{L}^{kn}, L^{uk}\}$ which is encoded by $\mathbf{L} := \{0, ..., N-1, N\}$. Each pixel can be predicted in our formulation as either one of the known or unknown classes. In the first case, each pixel must have a semantic label, while the cluster id is unnecessary. This is due to the fact that once the ith pixel is labelled with $s_i \in \mathbf{L}^{kn}$, its corresponding cluster id g_i is invalid (which is denoted by $void$). When the pixel is predicted as the unknown class, it can be clustered to g_i. Hence, the ith pixel with known classes (or unknown classes) can be assigned with $\mathbf{goss}_i = (s_i, void)$ (or $\mathbf{goss}_i = (N, g_i)$). In practice, a classification model can predict s_i, and g_i can be determined after the unknown pixels are clustered.

3.2 Evaluation metrics

Appropriate evaluation metrics are fundamental in driving the popularization of a new image segmentation task [43, 46, 47]. In this subsection, we briefly review some popular existing metrics for relevant segmentation tasks and then introduce a metric tailored for the proposed GOSS.

Previous Metrics. Standard metrics for OSS include the false positive rate at 95% true positive rate (FPR at 95% TPR), the area under the receiver operating characteristics (AUROC)

[48, 49], and the area under the precision recall (AUPR) [50, 51]. Such metrics assess performance based on the overlap of anomaly score distributions between the known and unknown classes. However, they are not suited for evaluating GOSS since they do not need to classify the input as a known or unknown class, as GOSS requires each input pixel to be explicitly classified as belonging to a known or unknown class. Instead, GOSS requires each input pixel to be explicitly classified as belonging to a known or unknown class. Well-known metrics for GS include the variation of information [52], probabilistic rand index [53], F-measure [54], and segmentation covering [21]. These metrics are initially proposed to evaluate data clustering or edge detection quality. As no multi-class semantic labels are considered, they cannot be directly used to measure the performance of joint GS and OSS.

GOSS Quality We borrow the idea of the segment matching from the panoptic quality (PQ) [43] in panoptic segmentation (PS) and adapt the panoptic quality as GOSS quality to be suitable for evaluating GOSS. As shown in Fig. 1d, the GOSS output can be viewed as a set of predicted segments, which is similar to the panoptic output of PS. The primary distinction between GOSS and panoptic predictions lies in the capability of GOSS to predict unknown segments.

We treat the unknown pixels as a new class in addition to N known classes. Thus, there is a total of $N+1$ classes of segmentation. Utilizing segment matching, a predicted segment from GOSS is matched with corresponding ground truth segment when their Intersection over Union (IoU) is higher than 0.5. As illustrated in Fig. 2, this approach enables the identification of true positives (TP), false positives (FP), and false negatives (FN) for the predicted segments generated by GOSS. We let GQ^{kn} be the average GOSS quality over N known classes. Accordingly, GQ^{uk} is the GOSS quality of the unknown class:

$$\mathrm{GQ}^{kn} = \frac{1}{N} \sum_{j \in \mathbf{L}^{kn}} \frac{\sum_{(u,\hat{u}) \in \mathrm{TP}_j^{kn}} \mathrm{IoU}(u, \hat{u})}{\mathrm{TP}_j^{kn} + \frac{1}{2}\mathrm{FP}_j^{kn} + \frac{1}{2}\mathrm{FN}_j^{kn}} \tag{1}$$

$$\mathrm{GQ}^{uk} = \frac{\sum_{(u,\hat{u}) \in \mathrm{TP}^{uk}} \mathrm{IoU}(u, \hat{u})}{\mathrm{TP}^{uk} + \frac{1}{2}\mathrm{FP}^{uk} + \frac{1}{2}\mathrm{FN}^{uk}} \tag{2}$$

where $\mathrm{IoU}(u, \hat{u})$ calculates the Intersection over Union value for the predicted segment u and the ground-truth segment $\hat{u}$. Furthermore, TP_j^{kn}, FP_j^{kn}, and FN_j^{kn} denote true positives, false positives, and false negatives for the jth known class, respectively. Similarly, GQ^{uk} is obtained specially for the unknown class with its true positives TP^{uk}, false positives FP^{uk}, and false negatives FN^{uk}.

The metrics GQ^{kn} and GQ^{uk} are computed based on the GOSS output (see Fig. 1d). The known and unknown segments on the GOSS prediction are evaluated separately via GQ^{kn} and GQ^{uk}. However, a unified metric is required to simplify the evaluation. Thus, we define a metric GOSS Quality (GQ) as:

$$\mathrm{GQ} = \lambda \cdot \mathrm{GQ}^{kn} + (1-\lambda) \cdot \mathrm{GQ}^{uk} \tag{3}$$

where we set λ as the most natural number, 0.5, throughout the paper. If we simply average GQ over $N+1$ classes, then the ratio between known and unknown would be significantly biased ($N:1$). In Eq. (3), GQ takes care of the known and unknown segments equally. We also introduce GQ^{clu}, which only assesses the pixel clustering quality regardless of pixel class (see Fig. 1b). Refer to the supplementary material for more details of GQ^{clu}.

3.3 Challenges

The endeavour of aggregating pixels in an image into clusters, as necessitated by GOSS, poses greater challenges compared to the conventional OSS task. This increased complexity stems from the clustering of objects belonging to unknown classes, which substantially heightens the difficulty of the task.

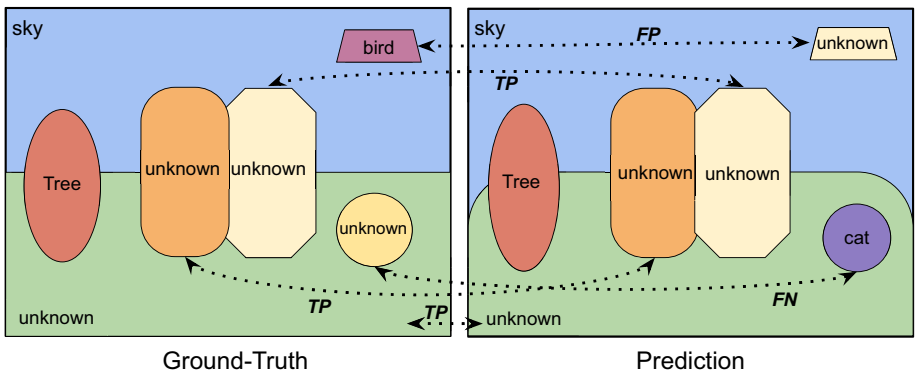


Fig. 2 Toy model of ground truth and predicted GOSS of an image. The predicted segments for "unknown" are partitioned into true positives TP^{uk}, false positives FP^{uk}, and false negatives FN^{uk}

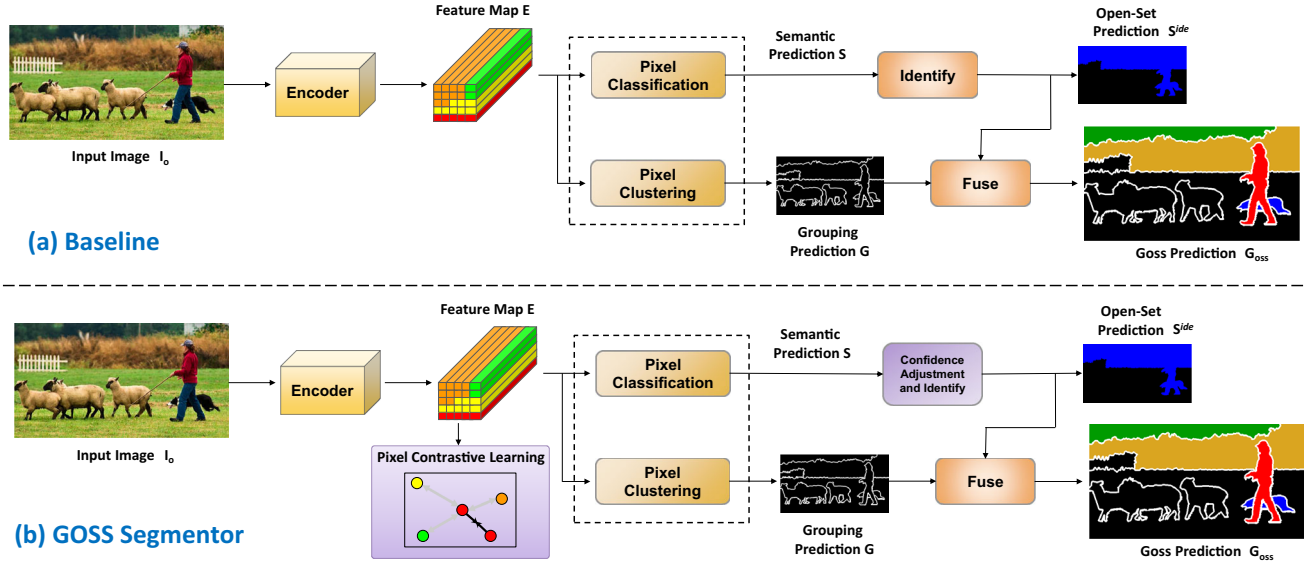


Fig. 3 The framework of the baseline and GOSS Segmentor. **a** Baseline. The input image is fed into the encoder for feature extraction. The dual-branch heads are jointly trained for pixel classification and clustering. Furthermore, pixel-wise contrastive learning is leveraged to learn discriminative feature embeddings. The pixel identification module is designed to recognize sets of pixels of the unknown class from the semantic prediction. The final GOSS output is generated by fusing the identified semantic and grouping predictions. **b** GOSS Segmentor (GST). Confidence adjustment and pixel contrastive learning modules are included

4 Methodology

In order to effectively perform GOSS, we propose a baseline framework (see Fig. 3a). The baseline is mainly comprised of five components: the shared encoder, the pixel classification branch, the pixel clustering branch, the identification module, and the fusion module. Then, we extend the baseline into a more advanced one, GOSS Segmentor (GST), as shown in Fig. 3b. More details of our design will be described next.

4.1 Baseline

GOSS can be modelled as a unified segmentation task incorporating pixel classification and clustering in an open-setting scenario. Given an image $I_o \in \mathbb{R}^{3\times\hbar_o\times\omega_o}$, we expect the proposed baseline to generate semantic and grouping predictions simultaneously. Hence, we adopt a dual-branch architecture, with one branch for pixel classification and another for pixel clustering. As shown in Fig. 3a, two branches share the same encoder as a feature extractor. The branch for pixel classification computes a prediction map $\mathbf{S} \in \mathbb{R}^{\hbar_o\times\omega_o}$, while the pixel clustering branch outputs a mask map $\mathbf{G} \in \mathbb{R}^{\hbar_o\times\omega_o}$ which includes the grouped class-agnostic segments. The unknown regions in $\mathbf{S}$ are identified, denoted by $\mathbf{S}^{ide}$, which is further fused with $\mathbf{G}$, obtaining the final GOSS output $\mathbf{G}_{oss} \in \mathbb{R}^{2\times\hbar_o\times\omega_o}$.

The baseline model is jointly trained with two losses: the classification loss ℓ_{cla} and the clustering loss ℓ_{clu}. The total loss is $\ell_{ws} = \alpha_{cla}\ell_{cla} + \alpha_{clu}\ell_{clu}$ where α_{cla} and α_{clu} are positive adjustment weights.

4.1.1 Pixel classification

We train the branch for pixel classification to classify each pixel as one of N classes where N is the number of predefined known classes. DeepLabV3+ [4], an existing powerful baseline for semantic segmentation, is chosen as the basic architecture for this branch. The branch is updated under ℓ_{cla} which is the cross-entropy loss between the predicted semantic map $\mathbf{S}$ and its ground-truth map.

DeepLabV3+ leverages an encoder–decoder architecture that takes a bottom-up pathway network with features at multiple spatial resolutions and appends a top-down pathway with lateral connections. The top-down pathway progressively upsamples features starting from the deepest layer of the network while concatenating or adding them with higher-resolution features from the bottom-up pathway. The Atrous Spatial Pyramid Pooling (ASPP) layer [3] is employed in the DeepLabV3+ model to enlarge the receptive field.

4.1.2 Pixel identification

Pixel identification from OSS is executed to identify sets of pixels of unknown classes from the semantic prediction. Hereafter, we study several recipes for pixel identification with which the identified semantic prediction $\mathbf{S}^{ide} \in \mathbb{R}^{\hbar_o\times\omega_o}$ is computed after processing $\mathbf{S}$. Common metrics like AUROC and AUPR of OSS assess the distribution overlap

between known and unknown classes. Still, pixel identification is required to state if the input pixel is known or not clearly. In other words, binary classification is necessary.

N-model In the pixel classification branch it is natural to design the semantic segmentation model with an N-dimensional confidence output $\mathbf{C} \in \mathbb{R}^{N\times\hbar_o\times\omega_o}$. The N-model is restricted to recognizing the set of predetermined known classes. When an unknown region comes up in a test image, it would be erroneously classified as one of the known classes. To identify unknown pixels based on outputs from the N-model, we employ the comparable OSS method, Maximum Softmax Probability (MSP) [55] or Maximum Unnormalized Logit (MaxLogit) [12]. Thresholds are used to classify pixels as belonging to a known or unknown class. More details can be found in the supplementary material.

N+1-model As opposed to the N-model, the N+1-model [56, 57] contains the unknown class in the output $\mathbf{C} \in \mathbb{R}^{(N+1)\times\hbar_o\times\omega_o}$ such that the N+1-model can directly identify the unknown pixels. During the training stage, the N+1-model explicitly takes the "unlabelled" pixels (i.e. the "void" pixels) as the unknown pixels. N+1-model is not valid if no "void" pixels are provided.

4.1.3 Pixel clustering

The pixel clustering branch is built in parallel with the pixel classification branch. The goal of this branch is to partition the whole image into clusters. During training, to generate the corresponding annotations, we convert the semantic labelling of the known classes into connectivity labelling by ignoring the previous semantics of each segment. The top-performing method, super boundary-to-pixel direction (Super-BPD) [23] is selected to establish the branch. The branch is trained in a supervised manner as well. Super-BPD applies the ground-truth annotation generated by the distance transform algorithm. Using the Super-BPD model, the representation of boundary-to-pixel direction (BPD) for each pixel is learned ($\ell_{clu} = \ell_{bpd}$). Super-BPDs are extracted based on the initial BPDs using the component-tree computation, followed by graph partitioning to merge super-BPDs into new segments.

4.1.4 Fusion module

The identification module outputs the identified semantic prediction $\mathbf{S}^{ide}$. Based on $\mathbf{S}^{ide}$, the grouping output $\mathbf{G}$ becomes $\mathbf{G}^{uk} \in \mathbb{R}^{\hbar_o\times\omega_o}$ where the element $g_i \rightarrow void$ if the corresponding semantic prediction $s_i \in [0, ..., N-1]$. Afterward, $\mathbf{S}^{ide}$ is merged with $\mathbf{G}^{uk}$ to form the GOSS output $\mathbf{G}_{oss} = [\mathbf{goss}_1, \mathbf{goss}_2, ..., \mathbf{goss}_{\hbar_o\omega_o}]$, where $\mathbf{goss}_i = (s_i, g_i) \in [0, ..., N] \times [1, ..., g_{max}] \cup void$. For ith pixel in $\mathbf{G}_{oss}$, $\mathbf{goss}_i = (s_i, void)$ if $s_i \in [0, 1, ..., N-1]$ or $\mathbf{goss}_i = (N, g_i)$ if $s_i = N$. The prediction $\mathbf{G}_{oss}$ can be viewed as a map that is composed of a set of several segments (see "GOSS prediction" in Fig. 3).

4.2 GOSS Segmentor

As shown in Fig. 3b, the baseline model is extended to a new model that we call GOSS Segmentor (GST). Keeping the original five baseline components, we propose to equip the baseline with a confidence adjustment module and a contrastive learning module.

4.2.1 Confidence adjustment

For the N+1 model, it is hard to accommodate unknown classes of objects since the model is trained without seeing any examples from these classes. Instead of completely trusting the confidence prediction $\mathbf{C}$, specific to the pixel identification of the N+1-model, we propose to modify $\mathbf{C}$ using a confidence adjustment. Particularly, for the ith pixel, its confidence score after softmax, $\mathbf{c}_i = [\mathbf{c}_i^{kn}, c_i^{uk}] \in \mathbb{R}^{N+1}$, is re-scaled as $[\mathbf{c}_i^{kn}, \beta^{uk} c_i^{uk}]$ where $\beta^{uk} \in (1, +\infty)$ is the scale coefficient of the confidence of the unknown class.

4.2.2 Pixel contrastive learning

In order to learn more discriminative representations for a better GOSS performance, inspired by [58], we adopt a pixel-wise contrastive learning algorithm where we contrast embeddings with different semantic labels. We have ith pixel embedding $\mathbf{e}_i \in \mathbb{R}^{cn}$ in the feature map $\mathbf{E} \in \mathbb{R}^{cn\times\hbar_o\times\omega_o}$ where cn, $\hbar_o$, and ω_o are the channel number, height, and width of the feature map. For $\mathbf{e}_i$, the positive pixel embeddings $\mathbf{e}_i^+$ have the same ground truth label to $\mathbf{e}_i$ in the same feature map, while the negative pixel embeddings $\mathbf{e}_i^-$ have different ground truth from $\mathbf{e}_i$. The pixel to pixel contrastive loss [58, 59] is then defined as:

$$\ell_{pc,i} = \frac{1}{|N_{pst,i}|} \sum_{i^+\in N_{pst,i}} -\log \frac{\exp(\mathbf{e}_i \cdot \mathbf{e}_i^+/\tau)}{\exp(\mathbf{e}_i \cdot \mathbf{e}_i^+/\tau) + \sum_{i^-\in N_{neg,i}} \exp(\mathbf{e}_i \cdot \mathbf{e}_i^-/\tau)} \tag{4}$$

where $N_{pst,i}$ and $N_{neg,i}$ are positive and negative embedding sets for pixel embedding $\mathbf{e}_i$. τ is the temperature parameter. We employ the semi-hard example sampling strategy from [58] to construct the positive and negative sample sets. Before the ℓ_{pc} is calculated, we downscale the ground-truth map to make it have the same size as the feature map $\mathbf{E}$.

The total loss ℓ_{ws} of GST is obtained by merging the pixel contrastive loss ℓ_{pc} with the classification loss ℓ^{cla} and clustering loss ℓ^{clu} as follows: $\ell_{ws} = \alpha_{cla}\ell_{cla} + \alpha_{clu}\ell_{clu} + \alpha_{pc}\ell_{pc}$, where α_{pc} is a positive adjustment weight for ℓ_{pc}. Contrastive learning aims to make the representations of pixels in the latent space closer when they belong to the same class and farther apart when they belong to different classes. This metric learning technique has been widely used in segmentation tasks, with many existing works reporting better empirical results [60–63]. In GOSS, we apply contrastive learning to train our model, with the aim of generating more representative embeddings for open-set evaluation.

5 Benchmark

Most datasets for OSS, like StreetHazards [12] and Road Anomaly [13], present separate unknown objects in an image. Ensuring that objects of unknown classes naturally appear together (are adjacent) in the image, in this work, we split the full set of labelling categories into known and unknown classes using a proper ratio. We simulate the training and testing of GOSS using existing semantic segmentation datasets, i.e. COCO-Stuff [11], and Cityscapes [9]. Note that grouping labels of unknown segments are derived from their initial ground-truth semantic labels before the split. Following [23], we convert the original semantic labelling of unknown areas to GS ground truths using connectivity labelling.

5.1 COCO-Stuff-GOSS

COCO-Stuff [11] augments the popular COCO [64] dataset with stuff classes as well as dense-pixel annotations. It has a large-scale semantic multi-class setting containing both the "things" and "stuff" classes. On COCO-Stuff, around 94% of the pixels are labelled with one semantic category, and the remaining are "unlabelled" pixels. We customize COCO-Stuff, creating a new benchmark named COCO-Stuff-GOSS. We strictly divide existing specific classes of COCO-Stuff into known and unknown classes. Training and testing images are selected from "train2017" and "val2017". Those categories which have been defined as unknown categories will not be represented in the training examples. Every selected testing example is composed of objects from the set of known categories and the set of unknown categories (or from only unknown categories). The statistics of the benchmark on different splits are shown in Table 2.

VOC Split The "VOC Split" is a common category split [65–68] that provides 20 "thing" classes defined in PASCAL VOC [47] as "known thing" classes. The remaining 60 "thing" classes are chosen as "unknown thing" classes.

Manual Split We divide COCO-Stuff categories according to how frequently each specific class appears. We count the number of occurrences of each class and calculate its ratio over the number of all training images. For example, in the "Manual-20/60" split, following that at least one and at most two classes are chosen from each sub-class, we select 20 of the most popular "thing" classes and treat the remaining "thing" classes as unknown. Besides, all "stuff" classes are set as known classes.

Random Split We also conduct experiments with a "Random Split", where all classes are randomly re-defined into known and unknown classes regardless of their super-class and sub-class. The data split of VOC-20/60 and Manual-20/60 does not include "stuff" categories as unknown classes, but Random-111/60 ensures that the known (or unknown) class includes specific classes from both the 'thing' and 'stuff' super-class. More details can be found in Table 2.

5.2 Cityscapes-GOSS

The Cityscapes [9] dataset consists of 5000 images (2975 train, 500 val, 1525 test) covering urban street scenes in driving scenarios. Dense pixel annotations of 19 classes are provided, that is, 8 "thing" and 11 "stuff" classes. As one goal of the proposed GOSS is to advance self-driving systems, we construct the Cityscapes-GOSS Benchmark. We divide the categories under the "manual split". As opposed to the COCO-Stuff-GOSS Benchmark, all images, regardless of containing unknown categories or not, are kept. We consider pixels from unknown classes as "void" pixels. Table 2 presents more details.

Manual Split We present two versions of the Cityscapes-GOSS Benchmark. Following the split in [14], we build the first version, "Manual-16/3", which includes "car", "truck", and "bus" as the "unknown thing". Based on the first version, we additionally make "building", "traffic sign", and "vegetation" as "unknown stuff" to produce a more challenging version, "Manual-13/6".

6 Experiment

Experimental results are presented in this section to demonstrate the rationality and effectiveness of GOSS. Using the baseline and proposed GST, we perform our task on COCO-Stuff-GOSS and Cityscapes-GOSS. The performance is mainly measured via the metric GQ.

6.1 Implementation

For all models, ResNet-50 [69] pre-trained on ImageNet [70] is utilized as the encoder backbone. All models are trained for

Table 2 Details of different splits of the COCO-Stuff-GOSS/Cityscapes-GOSS benchmarks. The numbers in the table indicate for each data split, how many known (or unknown) classes are selected and how many training (or testing) images are kept

Dataset	Data split	Known	Unknown	Known (thing)	Unknown (thing)	Known (stuff)	Unknown (stuff)	Train images	Test images
COCO-Stuff [11]	–	171	0	80	0	91	0	118287	5000
COCO-Stuff-GOSS	VOC-20/60	111	60	20	60	91	0	21711	4080
	Manual-20/60	111	60	20	60	91	0	17293	4264
	Random-111/60	111	60	51	29	60	31	18707	4156
Cityscapes [9]	–	19	0	8	0	11	0	2975	500
Cityscapes-GOSS	Manual-16/3	16	3	5	3	11	0	2975	500
	Manual-13/6	13	6	5	3	8	3	2975	500

We also provide details of the original COCO-Stuff [11] and Cityscapes [9] for comparison

60K/40K iterations with a batch size of 10/2 on COCO-Stuff-GOSS/Cityscapes-GOSS. The "poly" learning rate policy [71] is applied with the initial learning rate being set to 5e−5. GST models are updated using Adam optimization [72] without weight decay. The weights α_{cla}, α_{clu}, and α_{pc} are 1.0, 1e−4, and 1e−1. The thresholds in the identification module and the scale β^{uk} (for +CA) are set to 0.5 (0.75 for Cityscape-GOSS) and 5.0, respectively. Our models are implemented in PyTorch [73]. We note that the N+1-model cannot be used for the Cityscape-GOSS dataset. As we consider labels of unknown classes to be "void" instead of directly filtering out the image, the entropy of void pixels is not allowed to be added to the loss.

6.2 Results

The results of GOSS on COCO-Stuff-GOSS and Cityscapes-GOSS using various identification methods are reported in Tables 3 and 4, respectively. In addition to GOSS quality, we also provide metrics for OSS (AUROC and AURP) and GS (mIoU and GQ^{clu}) tasks to show that the models perform reasonably on these relevant older tasks.

For COCO-Stuff-GOSS in Table 3, GST becomes the best-performing model. For example, on the "Manual-20/60 split", GST attains 9.15% GQ, outperforming the N-model+MSP by a healthy margin of nearly 1.45%. Compared to other baselines, the pixel contrastive learning module assists GST to better discriminate between the known pixels and the unknown pixels in most cases (see "OSS Metric" in Table 3). Moreover, it boosts the clustering accuracy in GS. One of the baseline models, N-model+MaxLogit with using threshold, sacrifices much of GQ^{kn}, but it achieves a high GQ^{uk}. As expected, MaxLogit identifies more unknown areas. However, it does not simultaneously maintain the pixel classification accuracy [24]. For Cityscapes-GOSS in Table 4, we find a similar performance ranking to COCO-Stuff-GOSS in Table 3. For DML [14], except for the case on COCO-Stuff-GOSS of "VOC-20/60 split", it wins MSP and MaxLogit on the other benchmarks.

Several examples from the built benchmark are visualized in Fig. 4 to illustrate the GOSS setting better. Taking one example from Fig. 4 (2nd-row figure), GOSS accurately segments out "unknown dogs" from "unknown grass" (see Fig. 4f). Compared to the prediction of OSS in Fig. 4c, GOSS can provide richer information for intelligent agents to make decisions. With GOSS prediction, robots may avoid the obstacle ("unknown dogs") when they enter an unfamiliar scene ("unknown grass"). In terms of the GST model, we observe from Fig. 4 that the confidence adjustment module of GST effectively helps the N+1-model to detect more unknown regions (see Fig. 4b and c).

Table 3 GOSS results of GST (N+1-model+CA+CL) on COCO-Stuff-GOSS under three splits. "CA" is the confidence adjustment, and "CL" is the cross-pixel contrastive learning

Data split	Identification method	Clustering method	OSS metric		GS metric		GOSS metric		
			AUROC ↑	AUPR ↑	mIoU ↑	GQ^{clu} ↑	GQ^{kn} ↑	GQ^{uk} ↑	GQ ↑
VOC-20/60	N-model+MSP [55]	Super-BPD	76.5	26.4	12.7	27.3	14.4	3.0	8.70
	N-model+Maxlogit [12]		71.0	22.0	12.7	27.3	3.6	4.3	3.93
	N-model+DML [14]		71.1	20.9	12.5	26.6	1.6	4.3	2.93
	N+1-model [56]		75.7	25.7	12.4	25.9	14.3	3.3	8.83
	GST (ours)		77.0	27.0	13.6	28.7	15.4	4.9	**10.15**
Manual-20/60	N-model+MSP [55]	Super-BPD	74.7	27.8	12.8	26.1	13.1	2.3	7.70
	N-model+Maxlogit [12]		71.4	24.1	12.8	26.1	3.7	4.5	4.13
	N-model+DML [14]		75.9	28.5	12.8	27.3	13.3	2.3	7.81
	N+1-model [56]		76.2	29.5	12.9	27.4	13.1	2.2	7.62
	GST (ours)		77.3	30.7	13.8	28.3	14.3	3.9	**9.15**
Random-111/60	N-model+MSP [55]	Super-BPD	77.5	43.1	12.6	27.9	16.6	4.1	10.36
	N-model+Maxlogit [12]		73.9	41.2	12.6	27.9	3.7	8.7	6.22
	N-model+DML [14]		78.6	44.4	12.7	28.5	17.0	4.3	10.67
	N+1-model [56]		77.9	44.9	12.6	27.1	16.1	4.2	10.16
	GST (ours)		79.8	47.0	13.7	29.0	17.5	4.2	**10.87**

The best results of GQ are in bold

Table 4 GOSS results of GST (N-model+CL) on Cityscapes-GOSS under "Manual-16/3" and "Manual-13/6" splits

Data Split	Identification method	Clusteringmethod	GOSS Metric		
			GQ^{kn} ↑	GQ^{uk} ↑	GQ ↑
Manual-16/3	N-model+MSP [55]	Super-BPD	12.1	1.1	6.60
	N-model+Maxlogit [12]		6.2	1.7	3.92
	N-model+DML [14]		12.3	1.1	6.74
	GST (ours)		13.4	1.1	**7.31**
Manual-13/6	N-model+MSP [55]	Super-BPD	7.4	0.1	3.80
	N-model+Maxlogit [12]		3.4	0.6	2.01
	N-model+DML [14]		7.5	0.2	3.87
	GST (ours)		8.3	0.2	**4.26**

"CL" is cross-pixel contrastive learning. The best results of GQ are in bold

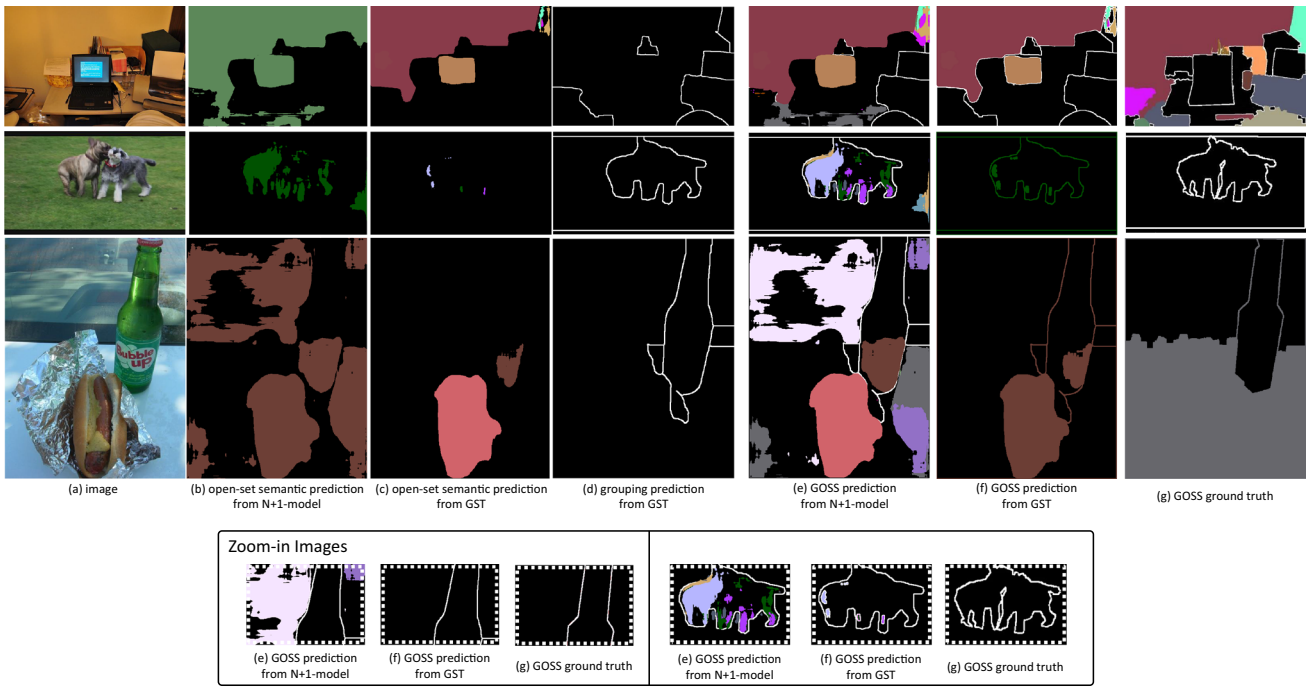


Fig. 4 Visualized segmentation results from GST (N+1-model+CA+CL) on COCO-Stuff-GOSS. The GOSS prediction (**f**) merges the OSS prediction (**c**) and the grouping prediction (**d**). Hence, within GOSS prediction (**f**), "objects" inside identified unknown regions can be segmented out. For example, unknown objects, "paper" in the 1st row, "dog/grass" in the 2nd row, and "bottle" in the 3rd row are correctly outlined even though their classes are not known. We also have some zoom-in images to show the effectiveness of the GST

6.3 Analysis

Number of Unknown Classes. The increased number of unknown classes makes the GOSS performance significantly drop as we observe the performance of "Manual-16/3" and "Manual-13/6" in Cityscapes GOSS (see Table 4).

Training Strategy. For all models in Table 3 and 4, the pixel classification branch and the pixel clustering branch are trained in a single unified architecture. Here, we study a different training strategy, "Separate," where two branches are trained separately, and their outputs are then merged. We find that the performances of "Separate" and our "Single" network are close. We finally choose "Single" since it is fast, light, and easy to implement.

Clustering Method. We train the pixel clustering branch for grouping unknown areas in an unsupervised manner by applying differentiable feature clustering (DFC) loss [74]. DFC in an unsupervised setting has a basic clustering performance, but it is worse than Super-BPD in a supervised setting.

Component Effect. Here, we discuss the effects of each model component ("CA" or "CL"). The results on COCO-Stuff-GOSS under the "Manual-20/60" split are shown in

Table 5 Ablation study on COCO-Stuff-GOSS under "Random-111/60" split: training strategy. Two strategies are compared

Data split	Identification method	Clustering method	Strategy	GOSS Metric		
				GQ^{kn} ↑	GQ^{uk} ↑	GQ ↑
Random-111/60	N-model+MSP [55]	Super-BPD	Single	16.6	4.1	10.36
	N-model+Maxlogit [12]			3.7	8.7	6.22
	N-model+MSP [55]		Separate	16.4	4.2	10.17
	N-model+Maxlogit [12]			3.4	8.9	6.14

Table 6 Ablation study on COCO-Stuff-GOSS under "Random-111/60" split: clustering method

Data Split	Identification method	Clustering method	GS Metric		GOSS Metric		
			mIoU ↑	GQ^{clu} ↑	GQ^{kn} ↑	GQ^{uk} ↑	GQ ↑
Random-111/60	N-model+MSP [55]	Super-BPD	12.6	27.9	16.6	4.1	10.36
	N-model+Maxlogit [12]		12.6	27.9	3.7	8.7	6.22
	N-model+MSP [55]	DFC	4.8	4.7	16.8	1.8	9.31
	N-model+Maxlogit [12]		4.8	4.7	3.7	3.6	3.69

Two clustering models of GST, super-BPD in a supervised setting and DFC [74] in an unsupervised setting are compared

Table 7 GOSS results of GST (N+1-model+CA+CL) on COCO-Stuff-GOSS under "Manual-20/60" split. "CA" is the confidence adjustment, and "CL" is the cross-pixel contrastive learning

Data split	Identification method	Clustering method	GOSS metric		
			GQ^{kn} ↑	GQ^{uk} ↑	GQ ↑
Manual-20/60	N+1-model	Super-BPD	13.1	2.2	7.62
	N+1-model+CA		14.0	2.5	8.24
	N+1-model+CL		14.1	3.5	8.82
	GST (Ours)		14.3	3.9	**9.15**

The best results of GQ are in bold

Table 7, which indicates "CA" or "CL" can solely boost the performance by a certain margin.

Challenging Task. The results in Table 3 and 4 verify that GOSS is a very challenging task, despite our baseline framework relying on strong backbones and a reasonable architecture. The first main reason is that it is non-trivial to perform accurate pixel identification under the open-set setting. For example, "unknown laptop" has been misclassified as "known tv" in the 1st-row figure in Fig. 4. Furthermore, the clustering branch suffers a performance drop when the model encounters the unfamiliar appearances of objects from unknown categories at test time. There is significant room for future improvement on the task of GOSS.

7 Conclusion

The improved setting referred to as GOSS is introduced in this paper. We aim to build upon the well-defined OSS to generate more comprehensive predictions. The task is to semantically classify pixels as one of the known classes or an unknown class and cluster the detected unknown pixels. With more extracted information inside the unknown region, GOSS might benefit intelligent agents in their decision-making process. Specific to the new setting, a metric, two benchmarks, and a corresponding baseline model are presented. In future works, the concept of GOSS can be further extended to include instance segmentation, image co-segmentation, video segmentation, point cloud segmentation, etc. We hope this work may provide a new alternative to a more comprehensive pixel-level scene understanding.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00371-023-02925-8.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions

Data and code availability The datasets generated and analysed during the current study are available in COCO (https://cocodataset.org) and Cityscapes (https://www.cityscapes-dataset.com) repositories.

Declarations

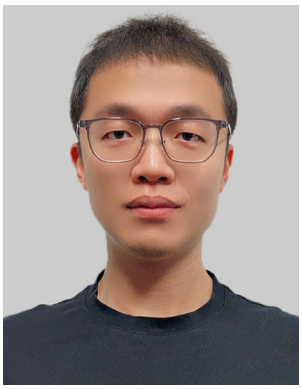
Conflict of interest The authors declare they have no conflict of interest.

References

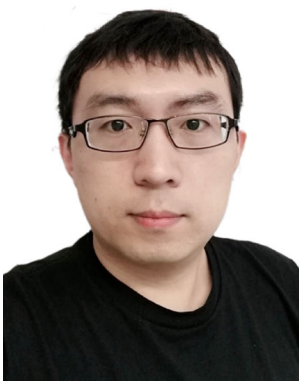
1. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
2. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal. Mach. Intell. **40**(4), 834–848 (2017)
3. Chen, L.-C., Papandreou, G., Schroff, F., Adam, H.: Rethinking Atrous Convolution for Semantic Image Segmentation. arXiv preprint arXiv:1706.05587 (2017)
4. Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 801–818 (2018)
5. Cheng, B., Schwing, A., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation. Adv. Neural Inform. Process. Syst. **34**, 17864 (2021)
6. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., *et al.*: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6881–6890 (2021)
7. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1290–1299 (2022)
8. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: European Conference on Computer Vision, pp. 1–15 (2006)
9. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3213–3223 (2016)
10. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 633–641 (2017)
11. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1209–1218 (2018)
12. Hendrycks, D., Basart, S., Mazeika, M., Zou, A., Kwon, J., Mostajabi, M., Steinhardt, J., Song, D.: Scaling out-of-distribution detection for real-world settings. In: Proceedings of the 39th International Conference on Machine Learning (ICML), pp. 8759–8773 (2022)
13. Lis, K., Nakka, K., Fua, P., Salzmann, M.: Detecting the unexpected via image resynthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2152–2161 (2019)
14. Cen, J., Yun, P., Cai, J., Wang, M.Y., Liu, M.: Deep metric learning for open world semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 15333–15342 (2021)
15. Yu, J., Kim, D.Y., Yoon, Y., Jeon, M.: Action matching network: open-set action recognition using spatio-temporal representation matching. Vis. Comput. **36**(7), 1457–1471 (2020)
16. Chan, R., Lis, K., Uhlemeyer, S., Blum, H., Honari, S., Siegwart, R., Salzmann, M., Fua, P., Rottmann, M.: Segmentmeifyoucan: A benchmark for anomaly segmentation. In: Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track (2021)
17. Bevandić, P., Krešo, I., Oršić, M., Šegvić, S.: Dense open-set recognition based on training with noisy negative images. Image Vision Comput. **124**, 104490 (2022)
18. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 888–905 (2000)
19. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. **24**(5), 603–619 (2002)
20. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. Int. J. Comput. Vision **59**(2), 167–181 (2004)
21. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 898–916 (2010)
22. Isaacs, O., Shayer, O., Lindenbaum, M.: Enhancing generic segmentation with learned region representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12946–12955 (2020)
23. Wan, J., Liu, Y., Wei, D., Bai, X., Xu, Y.: Super-bpd: Super boundary-to-pixel direction for fast image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9253–9262 (2020)
24. Jung, S., Lee, J., Gwak, D., Choi, S., Choo, J.: Standardized max logits: A simple yet effective approach for identifying unexpected road obstacles in urban-scene segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15425–15434 (2021)
25. Chan, R., Rottmann, M., Gottschalk, H.: Entropy maximization and meta classification for out-of-distribution detection in semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 5128–5137 (2021)
26. Xia, Y., Zhang, Y., Liu, F., Shen, W., Yuille, A.L.: Synthesize then compare: Detecting failures and anomalies for semantic segmentation. In: European Conference on Computer Vision, pp. 145–161 (2020). Springer
27. Di Biase, G., Blum, H., Siegwart, R., Cadena, C.: Pixel-wise anomaly detection in complex driving scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16918–16927 (2021)
28. Vojir, T., Šipka, T., Aljundi, R., Chumerin, N., Reino, D.O., Matas, J.: Road anomaly detection by partial image reconstruction with segmentation coupling. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15651–15660 (2021)
29. Kong, S., Ramanan, D.: Opengan: Open-set recognition via open data generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 813–822 (2021)
30. Szeliski, R.: Computer vision: algorithms and applications. Springer Nature (2022)
31. Grady, L.: Random walks for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **28**(11), 1768–1783 (2006)

32. Liu, Y., Jiang, P.-T., Petrosyan, V., Li, S.-J., Bian, J., 0001, L.Z., Cheng, M.-M.: Del: Deep embedding learning for efficient image segmentation. IJCAI 864, 870 (2018)
33. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, vol. 2, pp. 416–423 (2001). IEEE
34. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2650–2658 (2015)
35. Jafari, O.H., Groth, O., Kirillov, A., Yang, M.Y., Rother, C.: Analyzing modular cnn architectures for joint depth prediction and semantic segmentation. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 4620–4627 (2017). IEEE
36. Kokkinos, I.: Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6129–6138 (2017)
37. Bleyer, M., Rother, C., Kohli, P., Scharstein, D., Sinha, S.: Object stereo–joint stereo matching and object segmentation. In: CVPR 2011, pp. 3081–3088 (2011)
38. Yamaguchi, K., McAllester, D., Urtasun, R.: Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In: European Conference on Computer Vision, pp. 756–771 (2014). Springer
39. Sun, D., Sudderth, E.B., Black, M.J.: Layered segmentation and optical flow estimation over time. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1768–1775 (2012)
40. Sevilla-Lara, L., Sun, D., Jampani, V., Black, M.J.: Optical flow with semantic segmentation and localized layers. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3889–3898 (2016)
41. Hane, C., Zach, C., Cohen, A., Angst, R., Pollefeys, M.: Joint 3d scene reconstruction and class segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 97–104 (2013)
42. Hu, Y., Hugonot, J., Fua, P., Salzmann, M.: Segmentation-driven 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3385–3394 (2019)
43. Kirillov, A., He, K., Girshick, R., Rother, C., Dollár, P.: Panoptic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9404–9413 (2019)
44. Kirillov, A., Girshick, R., He, K., Dollár, P.: Panoptic feature pyramid networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6399–6408 (2019)
45. Hwang, J., Oh, S.W., Lee, J.-Y., Han, B.: Exemplar-based open-set panoptic segmentation network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1175–1184 (2021)
46. Cheng, B., Girshick, R., Dollár, P., Berg, A.C., Kirillov, A.: Boundary iou: Improving object-centric image segmentation evaluation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15334–15342 (2021)
47. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV (2010)
48. Davis, J., Goadrich, M.: The relationship between precision-recall and roc curves. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 233–240 (2006)
49. Fawcett, T.: An introduction to roc analysis. Pattern Recogn. Lett. **27**(8), 861–874 (2006)
50. Manning, C., Schutze, H.: Foundations of Statistical Natural Language Processing, (1999)
51. Saito, T., Rehmsmeier, M.: The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. PLoS ONE **10**(3), 0118432 (2015)
52. Meila, M.: Comparing clusterings: an axiomatic view. In: Proceedings of the 22nd International Conference on Machine Learning, pp. 577–584 (2005)
53. Rand, W.M.: Objective criteria for the evaluation of clustering methods. J. Am. Stat. Assoc. **66**(336), 846–850 (1971)
54. Martin, D.R., Fowlkes, C.C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans. Pattern Anal. Mach. Intell. **26**(5), 530–549 (2004)
55. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: International Conference on Learning Representations (ICLR) (2017)
56. DeVries, T., Taylor, G.W.: Learning Confidence for Out-of-Distribution Detection in Neural Networks. arXiv preprint arXiv:1802.04865 (2018)
57. Corbière, C., Thome, N., Bar-Hen, A., Cord, M., Pérez, P.: Addressing failure prediction by learning model confidence. In: Advances in Neural Information Processing Systems (NeurIPS) (2019)
58. Wang, W., Zhou, T., Yu, F., Dai, J., Konukoglu, E., Van Gool, L.: Exploring cross-image pixel contrast for semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 7303–7313 (2021)
59. Sohn, K.: Improved deep metric learning with multi-class n-pair loss objective. In: Advances in Neural Information Processing Systems, pp. 1857–1865 (2016)
60. Harley, A.W., Derpanis, K.G., Kokkinos, I.: Segmentation-aware convolutional networks using local attention masks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5038–5047 (2017)
61. De Brabandere, B., Neven, D., Van Gool, L.: Semantic instance segmentation with a discriminative loss function. arXiv preprint arXiv:1708.02551 (2017)
62. Hwang, J.-J., Yu, S.X., Shi, J., Collins, M.D., Yang, T.-J., Zhang, X., Chen, L.-C.: Segsort: Segmentation by discriminative sorting of segments. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
63. Zhao, S., Wang, Y., Yang, Z., Cai, D.: Region mutual information loss for semantic segmentation. arXiv preprint arXiv:1910.12037 (2019)
64. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European Conference on Computer Vision, pp. 740–755 (2014). Springer
65. O Pinheiro, P.O., Collobert, R., Dollar, P.: Learning to segment object candidates. In: Advances in neural information processing systems (NeurIPS) (2015)
66. Hu, R., Dollár, P., He, K., Darrell, T., Girshick, R.: Learning to segment every thing. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4233-4241 (2018)
67. Dhamija, A., Gunther, M., Ventura, J., Boult, T.: The overlooked elephant of object detection: Open set. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1021–1030 (2020)
68. Joseph, K., Khan, S., Khan, F.S., Balasubramanian, V.N.: Towards open world object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5830–5840 (2021)
69. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
70. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255 (2009). Ieee

71. Liu, W., Rabinovich, A., Berg, A.C.: Parsenet: Looking Wider to See Better. arXiv preprint arXiv:1506.04579 (2015)
72. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization. arXiv preprint arXiv:1412.6980 (2014)
73. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: an imperative style, high-performance deep learning library. Adv. Neural. Inf. Process. Syst. **32**, 8026–8037 (2019)
74. Kim, W., Kanezaki, A., Tanaka, M.: Unsupervised learning of image segmentation based on differentiable feature clustering. IEEE Trans. Image Process. **29**, 8055–8068 (2020)



Jie Hong received the M.Eng. degree from the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore, in 2017. He is currently pursuing the Ph.D. degree with the College of Engineering, Computing and Cybernetics, the Australian National University (ANU) and DATA61-CSIRO, Australia. His research interests include computer vision, deep learning, robotics and control systems.



Weihao Li is a Research Fellow at Australian National University (ANU), Australia. Before joining ANU in 2022, He was a Postdoctoral Research Fellow at Data61-CSIRO from 2019 to 2022. He received a PhD in Computer Science from Heidelberg University, Germany, under the supervision of Prof. Dr. Carsten Rother in 2019. His research interests are computer vision and machine learning.

Junlin Han earned his Bachelor of Information Technology (Honours) degree from the Australian National University (ANU) in 2023. His academic pursuits focus on computer vision, deep learning, and artificial intelligence, with a specific emphasis on leveraging data-centric methodologies to attain human-like visual intelligence. Junlin Han actively participates as a reviewer for esteemed conferences including CVPR, ICCV, ECCV, ICML, NeurIPS, and ICLR. He was honored as a NeurIPS top reviewer in 2022.

Jiyang Zheng received his bachelor's degree in computer science from The Australian National University (ANU) in 2022. He is currently a Ph.D. candidate at the Sydney AI Center (SAIC), University of Sydney. His research interest is trustworthy machine learning, with a particular focus on label-noise robustness and model generalisation.

Pengfei Fang is an Associate Professor at the School of Computer Science and Engineering, Southeast University (SEU), China. Before joining SEU, he was a post-doctoral fellow at Monash University in 2022. He received the Ph.D. degree from the Australian National University and DATA61-CSIRO in 2022, and the M.E. degree from the Australian National University (ANU) in 2017. His research interests include computer vision and machine learning.

Mehrtash Harandi is an Associate Professor with the Department of Electrical and Computer Systems Engineering at Monash University. He is also a contributing research scientist in the Machine Learning Research Group (MLRG) at Data61-CSIRO and an associated investigator at the Australian Center for Robotic Vision (ACRV). His current research interests include theoretical and computational methods in machine learning, computer vision, signal processing, and Riemannian geometry.

Lars Petersson is a Group Leader and Principal Research Scientist within the Imaging and Computer Vision Group, Data61, CSIRO, Australia. He is also leading one of the activities under CSIRO's Machine Learning and Artificial Intelligence Future Science Platform effort where data science problems from the smallest of microscopy scales to the largest of astronomical scales are addressed. Previous to joining Data61-CSIRO, he was a Principal Researcher and Research Leader in NICTA's computer vision research group where he was leading projects such as Smart Cars, AutoMap, and Distributed Large Scale Vision.