What's Outside the Intersection? Fine-grained Error Analysis for Semantic Segmentation Beyond IoU

Maximilian Bernhard[1,2], Roberto Amoroso[3], Yannic Kindermann[1]
Lorenzo Baraldi[3], Rita Cucchiara[3], Volker Tresp[1,2], Matthias Schubert[1,2]

[1]LMU Munich, [2]MCML, [3]University of Modena and Reggio Emilia

bernhard@dbs.ifi.lmu.de

Abstract

Semantic segmentation represents a fundamental task in computer vision with various application areas such as autonomous driving, medical imaging, or remote sensing. For evaluating and comparing semantic segmentation models, the mean intersection over union (mIoU) is currently the gold standard. However, while mIoU serves as a valuable benchmark, it does not offer insights into the types of errors incurred by a model. Moreover, different types of errors may have different impacts on downstream applications. To address this issue, we propose an intuitive method for the systematic categorization of errors, thereby enabling a fine-grained analysis of semantic segmentation models. Since we assign each erroneous pixel to precisely one error type, our method seamlessly extends the popular IoU-based evaluation by shedding more light on the false positive and false negative predictions. Our approach is model- and dataset-agnostic, as it does not rely on additional information besides the predicted and ground-truth segmentation masks. In our experiments, we demonstrate that our method accurately assesses model strengths and weaknesses on a quantitative basis, thus reducing the dependence on time-consuming qualitative model inspection. We analyze a variety of state-of-the-art semantic segmentation models, revealing systematic differences across various architectural paradigms. Exploiting the gained insights, we showcase that combining two models with complementary strengths in a straightforward way is sufficient to consistently improve mIoU, even for models setting the current state of the art on ADE20K. We release a toolkit for our evaluation method at https://github.com/mxbh/beyond-iou.

1. Introduction

Semantic segmentation is the task of assigning a semantic class label to each pixel in an image and is one of the most relevant problems in computer vision. There are plenty of corresponding application domains and datasets, all with different challenges and concerns. For instance, precise delineation of class boundaries may be critical for one application, such as medical image segmentation [41]. For another application, such as land use and land cover segmentation [16], however, clear-cut boundaries may not be available in the ground-truth annotations (or may not even exist), shifting the focus toward accurate classification, along with rather coarse localization. Furthermore, there are different learning settings in semantic segmentation, *e.g.*, weakly, semi-, or unsupervised, which also pose different challenges. For example, the correct segmentation of non-discriminative object parts is known to be a crucial problem in weakly supervised semantic segmentation and has attracted much attention from researchers [5, 21, 23, 24, 30].

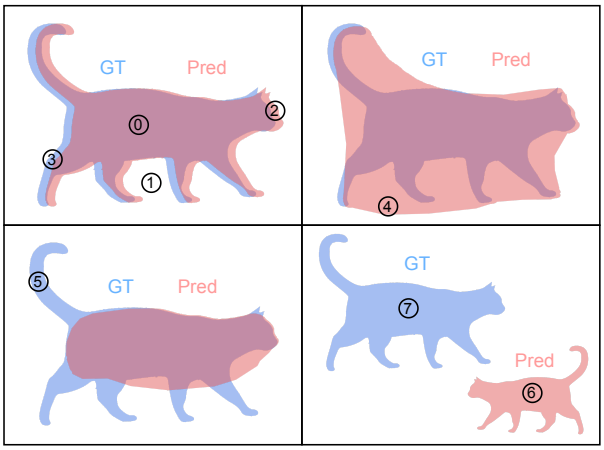

Figure 1. **Our proposed error categorization.** Each pixel is assigned to one of these categories: 0: true positive, 1: true negative, 2: false positive boundary, 3: false negative boundary, 4: false positive extent, 5: false negative extent, 6: false positive segment, 7: false negative segment.

Numerous semantic segmentation architectures and methods have been proposed to account for the diversity

of the task. Typically, these semantic segmentation methods and applications are evaluated via the *mean intersection over union ($mIoU$)*, which is currently the gold standard for model comparison and benchmarking. As the name already suggests, $mIoU$ measures the average intersection over union across all classes in a dataset. While it has the advantage of being an interpretable metric, it does not suit every application and dataset equally well. Moreover, $mIoU$ does not convey any insights into what types of errors a model makes. Other metrics that are occasionally used, *e.g.*, *F1-measure*, *Pixel Accuracy* or *Boundary IoU* [11], behave similarly and share this shortcoming. Thus, we argue that the current evaluation metrics, and especially $mIoU$ as a single metric, are not sufficient to evaluate semantic segmentation models in a differentiated way across various applications and datasets.

In this paper, we propose an intuitive error categorization that allows us to assess models w.r.t. various aspects, thereby meeting the diversity of semantic segmentation applications. We distinguish segmentation errors by assigning one of the following three categories (illustrated in Figure 1) to every incorrect pixel: (1) **Boundary errors** indicate that the model was able to correctly detect a transition between two semantic classes in the region of the respective pixel, but failed at the exact delineation of the boundary. (2) **Extent errors** indicate that a model recognized an instance (represented by a contiguous segment) and its class, but, in contrast to boundary errors, severely over- or underestimated its extent (*e.g.*, missing non-discriminative parts). (3) **Segment errors** are in no apparent relation to true positive predictions, *i.e.*, entire segments are mispredicted. Thus, a high number of segment errors indicates a model's weakness in the classification of its predicted segments. We propose that the mean error over union for each of these error types represents an intuitive extension of $mIoU$ and allows to determine how much loss in $mIoU$ each error type causes. Based on these error rates, conclusions about the model strengths and weaknesses can be drawn.

We validate our error categorization in extensive experiments comprising a detailed comparison of a variety of methods, datasets, and learning settings. In particular, we find that the advantage of the cutting-edge segmentation architectures MaskFormer [13], Mask2Former [12], and OneFormer [20] stems from a superior capability in precisely delineating boundaries and properly predicting segment extents. At the same time, these architectures have a remarkable weakness in the classification of segments. Leveraging this finding, we combine these models with other models that do not share this weakness and observe consistent improvements in $mIoU$.

In summary, our main contributions in this paper are: (i) We propose an intuitive error categorization allowing for a fine-grained analysis of semantic segmentation models. (ii) We validate our error categorization with extensive experiments on commonly used benchmark datasets as well as a sensitivity analysis under systematic errors. (iii) We use our analysis to perform a broad comparison of current semantic segmentation models, gaining valuable insights about different architectures. (iv) From these insights, we derive a straightforward way to improve the performance of Mask2Former and OneFormer, surpassing state-of-the-art results on ADE20K without training new models.

2. Background and Related Work

The most common metric for evaluating semantic segmentation models is $mIoU$. Apart from that, there are other metrics such as *F1-measure/Dice score* and *Pixel Accuracy*, which are occasionally considered in addition to $mIoU$ [11]. Furthermore, the per-image *BF score* has been proposed in [15], aiming to measure segmentation quality in a way that is close to human perception. *Trimap IoU* [7, 22] measures IoU restricted to a band of pixels along the boundaries of ground-truth masks. Thus, *Trimap IoU* is insensitive to errors far from ground-truth mask boundaries and favors larger predictions. In [11], the so-called *Boundary IoU* was proposed. It measures the intersection over union only on pixels that lie close to the boundaries of predicted or ground-truth foreground masks. Restricting the set of pixels for IoU computation to the boundary pixels has the advantage of mitigating IoU's bias toward large objects. However, at the same time, it introduces an insensitivity toward errors far away from predicted and ground-truth mask boundaries. For an overview and comparison of these evaluation metrics, we refer to [11].

Altogether, these metrics are designed to measure the overall performance of a model. To shed some light on the errors of a model, one can consider *Precision* and *Recall*, providing rather limited insights as errors are only distinguished in terms of false positives and false negatives. Alternatively, one can resort to a qualitative visual inspection. However, a qualitative analysis can be time-consuming and subject to a relatively high variance due to the limited number of samples that can be examined.

Therefore, a method that allows to assess and compare models in a fine-grained and quantitative manner would be beneficial for both researchers and practitioners. To the best of our knowledge, there is currently no such solution for semantic segmentation. For the task of object detection, however, there is a tool called TIDE [2], which breaks down detection errors and indicates the loss in *Average Precision (AP)* caused by the different error types. TIDE has proven to be useful for analyzing models and justifying certain design choices [6, 10, 25, 29, 38]. With this paper, we fill this gap for semantic segmentation, enabling a fine-grained, quantitative model evaluation.

3. Error Categorization

Overall, semantic segmentation can be seen as a per-pixel classification task, reflected by the evaluation via IoU, treating each incorrect pixel separately and identically. However, from a human perspective, segmentation errors can appear rather diverse, as our perception is focused on entire visual instances formed by segments and objects instead of single pixels. Thus, it is sensible to base an error categorization on the concept of contiguous segments, considering the relations between pixels. On a high level, we distinguish between erroneous pixels belonging to (at least partially) correctly predicted segments (boundary and extent errors) and pixels belonging to completely erroneous segments (segment errors). A visual overview is provided in Figure 1.

3.1. Notation and Preliminary Definitions

Like $mIoU$, our proposed categorization of errors considers all classes of a dataset separately, which is why we subsequently only consider a single class. That is, for an image with pixel locations $\Omega = [H]\times[W]$, we denote the binary ground-truth segmentation with $G \in \{0,1\}^{H\times W}$ and the corresponding binary prediction with $P \in \{0,1\}^{H\times W}$, *i.e.*, zero and one indicate background and foreground for this class, respectively. The true positive pixels are defined as $TP = \{x \in \Omega \mid G_x = 1 \wedge P_x = 1\}$ and FP, FN, and TN follow analogously. We define the d-neighborhood of a pixel as $\mathcal{N}_d(x) = \{x' \in \Omega \mid \delta(x,x') \leq d\}$, where $\delta(\cdot,\cdot)$ denotes the Euclidean pixel distance rounded to the nearest integer. Thus, $\mathcal{N}_1(x)$ describes x plus its eight neighboring pixels. Furthermore, we introduce an operator $\mathcal{S}(\cdot)$, that extracts all contiguous segments with label one from a binary segmentation mask such as G or P[1]. For ease of notation, we also allow a binary mask represented by a set of pixels describing the locations of ones as input to $\mathcal{S}(\cdot)$. Each contiguous segment $s \in \mathcal{S}(\cdot)$ is represented by a set of pixel locations, *i.e.*, $s \subseteq \Omega$. Moreover, let $\mathcal{S}(\cdot)_x$ denote the unique contiguous segment that contains pixel location x, given that the provided binary mask has label one at location x.

3.2. Boundary Errors

A boundary error in our categorization should occur when a transition between foreground and background for a class has been recognized, but not delineated perfectly. Thus, we first formulate a preliminary definition via the occurrence of true positive and true negative pixels in the neighborhood, *i.e.*,

$$\begin{aligned} FP'_{bnd} &= \{x \in FP \mid \mathcal{N}_d(x) \cap TP \neq \emptyset \wedge \mathcal{N}_d(x) \cap TN \neq \emptyset\} \\ FN'_{bnd} &= \{x \in FN \mid \mathcal{N}_d(x) \cap TP \neq \emptyset \wedge \mathcal{N}_d(x) \cap TN \neq \emptyset\} \\ E'_{bnd} &= FP'_{bnd} \cup FN'_{bnd}. \end{aligned} \tag{1}$$

According to this definition, boundary errors are not just prediction errors along the boundaries of the ground truth, but they require proximity to the boundaries of both the ground truth and the prediction. The dependency on the distance parameter d is discussed in Section 3.6 and suppressed in the notation of the error sets for readability.

In addition, we propose two modifications to this definition to account for unwanted effects. First, to avoid transitions between boundary and non-boundary errors (see Figure 2), we extend the boundary error area via

$$\begin{aligned} FP''_{bnd} &= FP'_{bnd} \cup \{x \in FP \mid \mathcal{N}_d(x) \cap FP'_{bnd} \neq \emptyset\} \\ FN''_{bnd},\ E''_{bnd} &\text{ analogous.} \end{aligned} \tag{2}$$

Thus, boundary errors can be at most $2d$ pixels away from true positive and true negative pixels.

Second, we remove all contiguous segments that have no true positives or no true negatives as direct neighbors, *i.e.*,

$$\begin{aligned} FP_{bnd} = \bigcup \{ s \in \mathcal{S}(FP''_{bnd}) \mid &\exists_{x_1 \in s} : \mathcal{N}_1(x_1) \cap TP \neq \emptyset \\ &\wedge \exists_{x_2 \in s} : \mathcal{N}_1(x_2) \cap TN \neq \emptyset \} \\ FN_{bnd},\ E_{bnd} &\text{ analogous.} \end{aligned} \tag{3}$$

Hence, if a contiguous segment of potential boundary errors $s \in \mathcal{S}(FP''_{bnd})$ is not adjacent to at least one true positive and one true negative, we do not regard its pixels as boundary errors as a correct transition between foreground and background of the class is not present in this case.

3.3. Extent Errors

Extent errors describe errors that occur when a segment has been recognized, but largely over- or underestimated in its extent (*e.g.*, when non-discriminative parts are not recognized). In the false positive case, they are pixels that belong to a contiguous predicted segment intersecting with the ground-truth foreground, and in the false negative case, there are pixels that belong to a contiguous ground-truth segment intersecting with the predicted foreground. Formally, we define extent errors as

$$\begin{aligned} FP_{ext} &= \{x \in FP \setminus FP_{bnd} \mid \mathcal{S}(P)_x \cap TP \neq \emptyset\} \\ FN_{ext} &= \{x \in FN \setminus FN_{bnd} \mid \mathcal{S}(G)_x \cap TP \neq \emptyset\} \\ E_{ext} &= FP_{ext} \cup FN_{ext} \end{aligned} \tag{4}$$

In other words, extent errors can be thought of as error pixels that would become boundary errors if the distance parameter d was increased to infinity. As extent errors can have an arbitrary distance to the boundary of their corresponding ground-truth segment, we consider them more severe than boundary errors.

[1] implemented with `scipy.ndimage.label`

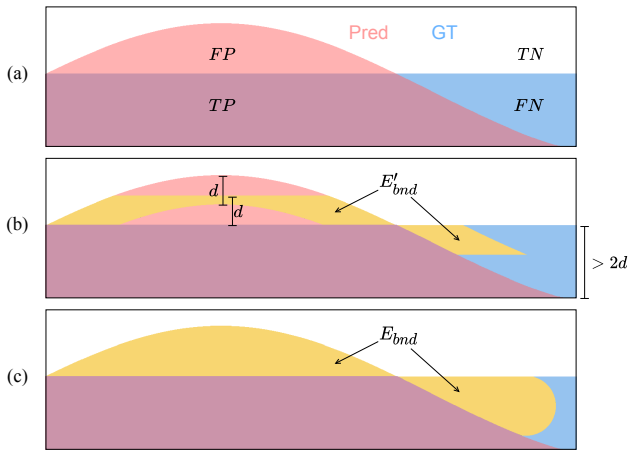

Figure 2. **Extending the boundary area** E'_{bnd} in subfigure (b) to E_{bnd} in subfigure (c) via Equation 2 avoids unwanted transitions in the form of $TP \rightarrow E_{ext} \rightarrow E'_{bnd} \rightarrow E_{ext} \rightarrow TN$.

3.4. Segment Errors

After having defined boundary and extent errors, we now come to errors that have no apparent relation to true positive predictions. That is, we are now dealing with predicted segments that do not have any intersection with the ground-truth foreground (false positive) and ground-truth foreground segments that do not have any intersection with the predicted foreground (false negative). We call these errors segment errors and define them as

$$\begin{aligned} FP_{seg} &= \{x \in FP \mid \mathcal{S}(P)_x \cap TP = \emptyset\} \\ FN_{seg} &= \{x \in FN \mid \mathcal{S}(G)_x \cap TP = \emptyset\} \\ E_{seg} &= FP_{seg} \cup FN_{seg}. \end{aligned} \tag{5}$$

Therefore, segment errors occur when models predict wrong classes for entire segments, and a large number of segment errors indicates poor performance in classification.

3.5. Error Statistics

Error over Union Since the proposed error categorization assigns each false positive and false negative pixel to exactly one error category, we can count the number of pixels for each error category and all images and define the error over union analogously to IoU, *i.e.*,

$$E_\star oU = \frac{|E_\star|}{|U|}, \star \in \{bnd,\ ext,\ seg\}, \tag{6}$$

where $U = TP \cup FP \cup FN$. Furthermore, we define $mE_{\{bnd,ext,seg\}}oU$ analogously to $mIoU$ as the mean error over union over all classes. As the three error categories are disjoint, the IoU plus the sum of all errors over union yields a total of one,

$$IoU + E_{bnd}oU + E_{ext}oU + E_{seg}oU = 100\%. \tag{7}$$

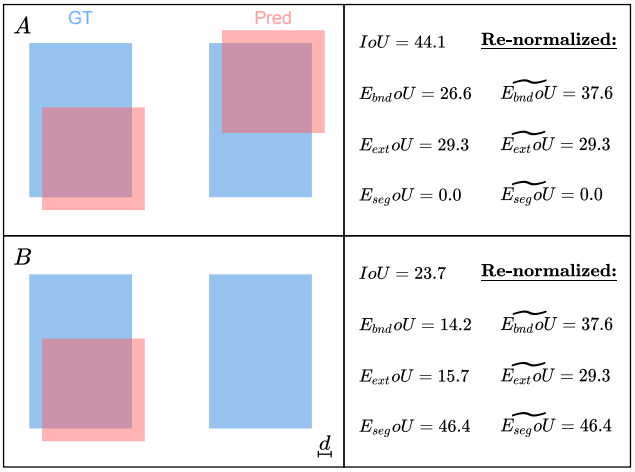

Figure 3. **The effect of re-normalizing $E_\star oU$.** A (top) and B (bottom) are two models equally good at segmenting boundaries and extents. However, A has substantially higher values for $E_{bnd}oU$ and $E_{ext}oU$ than B because B completely misses the second ground-truth segment. The re-normalized error over union accounts for this effect such that A and B solely differ in $\widetilde{E_{seg}oU}$, while $\widetilde{E_{bnd}oU}$ and $\widetilde{E_{ext}oU}$ are identical.

Clearly, the same holds for $mIoU$ and $mE_{\{bnd,ext,seg\}}oU$. Therefore, the error over union quantifies how much loss in $mIoU$ each error type causes. This kind of interpretability makes the proposed error categorization easy to grasp and perfectly fit with the evaluation via $mIoU$.

Re-normalized Error over Union However, when comparing models with rather different strengths, it is sensible to consider another quantity next to $mIoU$ and $mE_{\{bnd,ext,seg\}}oU$ to gain reliable insights.

Consider a model A, which has better classification capabilities and, therefore, a lower $E_{seg}oU$ than another model B. Due to the fewer segment errors, model A will face more occasions to produce boundary and extent errors. Hence, the errors over union for boundary and extent would show larger values for model A, even if A and B had equal performances in these regards. Carrying this logic forward, a lower $E_{ext}oU$ will cause larger values for $E_{bnd}oU$. To account for this, we propose the re-normalized errors over union

$$\begin{aligned} \widetilde{E_{seg}oU} &= E_{seg}oU \\ \widetilde{E_{ext}oU} &= \frac{|E_{ext}|}{|U| - |E_{seg}|} = \frac{|E_{ext}|}{|TP| + |E_{bnd}| + |E_{ext}|} \\ \widetilde{E_{bnd}oU} &= \frac{|E_{bnd}|}{|U| - |E_{seg}| - |E_{ext}|} = \frac{|E_{bnd}|}{|TP| + |E_{bnd}|}. \end{aligned} \tag{8}$$

That is, for each error type, we remove the more fundamental errors (in terms of localization) from the denominator. We can interpret the re-normalized error over union as

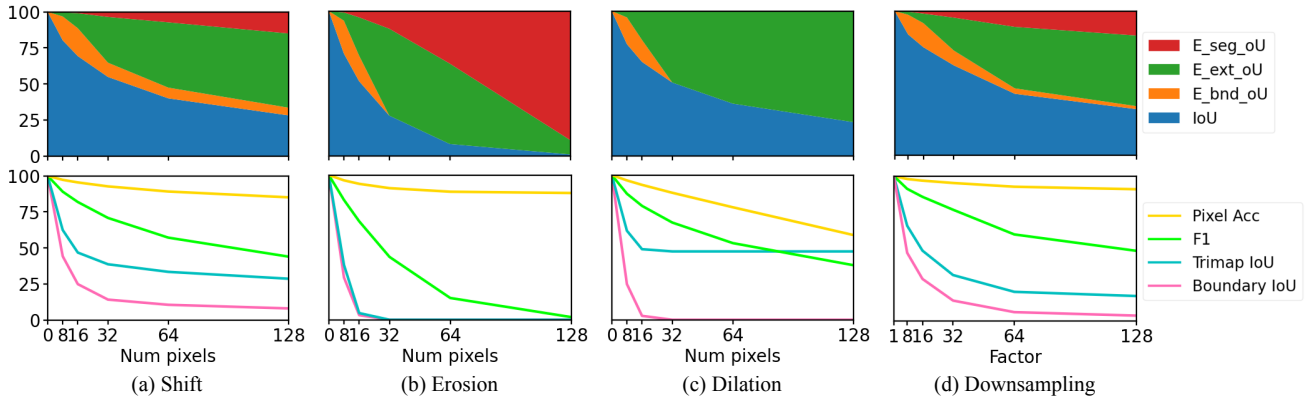

Figure 4. **Sensitivity analysis** for our error types (top) and existing metrics (bottom) under different transformations.

the loss in IoU caused by an error type if the model had perfect performance w.r.t. to the other error types. The re-normalized error over union still ranges from zero to one; however, it loses the property that the IoU plus the error rates sum up to one. An illustration of the effect of the re-normalization is provided in Figure 3.

3.6. Choosing the Distance Parameter d

For choosing the parameter d, similar considerations as for the pixel distance parameter in Boundary IoU [11] apply. In our error categorization, d determines how far away from the true boundary a boundary error may occur (at most $2d$). Decreasing d will lead to fewer boundary and more extent errors, whereas setting d loosely will increase the number of boundary errors and small or thin segments may only consist of their boundaries, thereby reducing the number of extent errors. Overall, d should be chosen such that a deviation of $2d$ can still be considered close to the true boundary for the application and the dataset at hand.

Like [11], we set d dependent on the image size. For ADE20K [40], PascalVOC [17], COCO-Stuff 164k [3], and STARE [19], we select d as 1% of the image diagonal, whereas for CityScapes [14] and iSAID [32], we use 0.25% and 0.5% of the image diagonal, respectively, due to their high-resolution images and high-quality annotations.

4. Experiments

4.1. Sensitivity Analysis

To facilitate a better understanding of our proposed error categorization, we analyze the sensitivity of the different error types toward systematic transformations of ground-truth masks. That is, we take binary ground-truth masks from ADE20K, corrupt them, and compute the error rates compared to the original ground truth. For the corruptions, we use shifting, erosion, dilation, and downsampling of different severities to simulate various erroneous prediction behaviours. In particular, with downsampling, we mimic the predictions of a model that cannot produce precise delineations at class boundaries. To have well-defined segmentation masks with exactly one class label for each pixel after applying the transformations, we consider only binary masks and compute the metrics for a single global foreground class.

Looking at the results in Figure 4 (a), we see that small shifts increase boundary errors. If the number of pixels by which the masks are shifted increases further, extent errors go up first before some segments do not overlap with their original ones anymore, additionally leading to segment errors. For erosion and dilation (see Figure 4 (b,c)), we observe a similar behavior for boundary errors under light corruptions. Also, extent errors grow rapidly if the number of erosion/dilation pixels is further increased. However, in this case, boundary errors are completely replaced by extent errors as erosion and dilation inevitably move boundaries away from their original locations. Furthermore, we can observe segment errors for erosion with high severity, as small segments may completely vanish with erosion, whereas this cannot happen with dilation. Similar to shift, erosion, and dilation, the downsampling corruption in Figure 4 (d) also leads to an increase in boundary errors for small severities. When the downsampling factor is increased to rather extreme values of 64 and 128, we observe segment errors because small segments are lost at such low resolutions.

On the other hand, if we look at existing evaluation metrics in the bottom row of Figure 4, we observe monotonically decreasing curves as these metrics are primarily designed to measure the overall segmentation quality. However, this makes it hard to gain insight into prediction errors. In contrast, the error rates for our proposed categories reach their maxima at different severities and behave differently under different corruptions, making it easier to develop an understanding of the prediction errors. To sum up, this analysis of error types under systematic corruption

	Architecture	Backbone	mIoU	mE$_\star$oU			$\widetilde{\text{mE}_\star\text{oU}}$		
				bnd	*ext*	*seg*	*bnd*	*ext*	*seg*
(1)	PSPNet [37]	R101-D8	44.4	9.0	15.9	30.7	19.2	25.2	30.7
(2)	DeepLabV3 [8]	R101-D8	45.0	9.5	15.9	29.6	20.0	24.9	29.6
(3)	DeepLabV3+ [9]	R101-D8	45.5	9.0	15.8	29.8	18.8	24.1	29.8
(4)	SETR [39]	ViT-L	48.3	10.1	15.1	26.6	19.5	22.0	26.6
(5)	SegFormer [34]	MiT-b5	49.6	9.6	13.0	27.8	18.1	19.9	27.8
(6)	Segmenter [28]	ViT-L	52.2	10.1	12.7	25.0	18.4	18.8	25.0
(7)	MaskFormer [13]	Swin-L	54.1	7.1	8.4	30.3	12.7	13.0	30.3
(8)	Mask2Former [12]	R101-D32	48.6	6.7	10.2	34.5	13.4	16.7	34.5
(9)	Mask2Former [12]	Swin-L	56.0	7.2	8.2	28.5	12.7	12.7	28.5
(10)	UPerNet [1, 33]	BEiT-L	56.3	8.4	11.7	23.6	14.5	17.1	23.6
(11)	OneFormer [20]	Swin-L	57.0	6.5	8.1	28.4	11.2	12.1	28.4
(12)	OneFormer [20]	DiNAT-L	58.0	6.6	7.9	27.6	11.0	11.4	27.6
(13)	UPerNet [33]	R50	42.1	8.9	17.5	31.5	20.1	28.1	31.5
(14)	UPerNet [33]	R101	43.8	8.7	16.0	31.5	19.3	25.8	31.5
(15)	UPerNet [33]	ViT-B	48.8	9.3	13.8	28.1	18.6	21.6	28.1
(16)	UPerNet [33]	Swin-B	50.8	9.1	13.1	27.1	17.0	19.8	27.1
(17)	UPerNet [33]	BEiT-B	53.1	8.8	13.8	24.3	16.0	19.6	24.3

Table 1. **Results of state-of-the-art models on ADE20K val.** CNN models are mostly outperformed by early transformer-based models w.r.t. extent and segment errors. Mask-classification based models (7-9,11,12) reach new levels of $mIoU$ only because of large improvements w.r.t. boundary and extent errors.

shows that the proposed error categories behave as expected and provide additional information to existing metrics. This justifies our design, and for further substantiation with qualitative examples, we refer to the supplementary material.

4.2. Comparing State-of-the-art Models

In Table 1, we provide a broad comparison of state-of-the-art semantic segmentation models on ADE20K. The selected models can be broadly categorized as CNN-based, transformer-based, and mask-classification-based architectures. For comparing the different models, our main focus lies on the re-normalized errors over union $\widetilde{mE_\star oU}$, but we also report $mE_\star oU$ for completeness and to allow for a better examination of single models.

The CNN architectures (1) PSPNet [37], (2) DeepLabV3 [8], and (3) DeepLabV3+ [9] generally perform rather similarly w.r.t. to $mIoU$ and all error types. If we compare DeepLabV3 and DeepLabV3+ more closely, we can see that DeepLabV3+ outperforms DeepLabV3 mainly in terms of boundary errors (18.8 vs. 20.0% $\widetilde{mE_{bnd}oU}$). Interestingly, this observation is in line with the claim that DeepLabV3+ is able to predict more precise boundaries [9].

The transformer-based methods (4) SETR [39], (5) SegFormer [34], and (6) Segmenter [28], reach higher $mIoU$ scores than the considered CNN models (partly due to stronger backbones). These performance gains mainly stem from a reduction in extent and segment errors, which is intuitive considering transformers' superior capabilities in contextualization and global reasoning.

However, (7) MaskFormer [13] and its conceptual successors (8,9) Mask2Former [12] and (11,12) OneFormer [20] are able to outperform these transformer methods while showing a remarkably different distribution of errors. The paradigm of classifying entire masks instead of single pixels leads to substantially lower error rates for boundaries and extents. On the other hand, these models have relatively high numbers of segment errors, *e.g.*, 34.5% for (8) Mask2Former + R101-D32.

The only method in our comparison that does not follow this paradigm but that can rival these architectures with a similarly sized backbone is (10) UPerNet + BEiT-L [1, 33]. The main strength of this model seems to be classification, as shown by the lowest segment error rate in our analysis (23.6%). At the same time, it produces significantly more boundary and extent errors than the mask-classification-based models. Hence, there are fundamental differences in the predictions and errors of state-of-the-art models, which we further analyze in Section 4.3.

In addition, we compare different backbones within the context UPerNet (13-17) in the lower section of Table 1. Once again, we observe that transformer models outperform CNNs mostly w.r.t. extent and segment errors. Furthermore, stronger backbones reduce errors across all categories rather uniformly, without a distinct advantage favoring any particular type of error.

Segmentor		Classifier	mIoU (+Δ)
(8) Mask2Former	∘	(4) SETR	50.1 *(+1.5)*
(9) Mask2Former	∘	(6) Segmenter	56.4 *(+0.4)*
(9) Mask2Former	∘	(10) UP.+BEiT	56.9 *(+0.6)*
(11) OneFormer	∘	(10) UP.+BEiT	57.9 *(+0.9)*
(12) OneFormer	∘	(10) UP.+BEiT	58.6 *(+0.6)*
(4) SETR	∘	*(8) Mask2Former*†	48.0 *(–0.3)*

Table 2. **Combining models with complementary strengths** consistently improves performance on ADE20K val, even beyond the state of the art set by OneFormer [20]. $A \circ B$ denotes applying model A for segmentation after multi-label classification with B. †: Combining model weaknesses for comparison.

Dataset	mIoU	mE⋆oU *bnd*	*ext*	*seg*
ADE20K [40]	42.5	9.3	17.4	30.8
COCO-Stuff 164k [3]	40.5	6.0	15.6	37.8
PascalVOC 2012 [17]	77.3	3.6	11.2	8.0
CityScapes [14]	79.6	8.4	6.5	5.6
iSAID [32]	65.4	11.8	7.2	15.6
STARE† [19]	84.0	15.4	0.4	0.2

Table 3. **Comparing error rates across datasets** from different domains with PSPNet (R50-D8). The datasets exhibit significantly different error distributions. †: UNet-S5-D16 backbone.

4.3. Case Study: Combining Models with Complementary Strengths

Our evaluation of architectures in Table 1 has shown that Mask2Former and OneFormer produce many segment errors, and their overall strong performances mainly come from few boundary and extent errors. At the same time, architectures like Segmenter and UPernet + BEiT produce substantially fewer segment errors, and their performance bottlenecks are boundary and extent errors. Therefore, the question arises whether models with such complementary strengths can benefit from each other. To test this hypothesis, we evaluate simple combinations of two models, where one model is employed for multi-label image classification and another model produces segmentation masks for the predicted classes. More precisely, for each class that is deemed to be absent in the image by the first model (classifier), we set the predicted logits of the second model (segmentor) to minus infinity. We keep the way we combine models deliberately simple and dispense with more sophisticated techniques since the primary goal of this experiment is to demonstrate the practicability of insights gained through our error analysis.

The results for these combinations of models are provided in Table 2. First, we combine (8) Mask2Former + ResNet-101, having the highest segment error rate in Table 1 (34.5%), with (4) SETR, having a similar overall performance, but much fewer segment errors (26.6%). With the combination, we can increase Mask2Former's $mIoU$ from 48.6% to 50.1%, while decreasing its $mE_{seg}oU$ from 34.5% to 33.4%. Conversely, if we combine these two models the other way around, *i.e.*, using (8) Mask2Former for classification and (4) SETR for segmentation, their weaknesses are emphasized, leading to a drop in $mIoU$ (48.0%).

The observed improvement in performance is not unique to Mask2Former with the relatively weak ResNet backbone but can also be achieved with the stronger Swin-L backbone. Even when combining (9) Mask2Former + Swin-L with (6) Segmenter + ViT-L, which has a 3.8% lower $mIoU$, the combined $mIoU$ improves by 0.4%, reaching 56.4%. Finally, we combine both (11) OneFormer + Swin-L and (12) OneFormer + DiNAT-L with (10) UPerNet + BEiT-L, reaching $mIoU$s of 57.9% and 58.6%, respectively. This surpasses the previous state-of-the-art on ADE20K val, which was set by OneFormer (single-scale inference, no additional training data, see [20] for details).

In summary, these results show that segment errors are a major limiting factor for mask-classification-based models such as Mask2Former and OneFormer. This insight opens up an intriguing research direction for improving these models and further advancing the state of the art in semantic segmentation. For our error analysis methodology, this case study is a clear demonstration of its usefulness for both researchers and practitioners.

4.4. Comparing Datasets

Since there are not only a large number of model architectures for semantic segmentation but also a variety of commonly used benchmark datasets coming from different application domains, it is worth looking into error rates on different datasets as well. In Table 3, we compare error rates of PSPNet on ADE20K [40], CityScapes [14], PascalVOC 2012 [17], COCO-Stuff 164k [3], iSAID [32], and STARE [19]. ADE20K, PascalVOC, and COCO-Stuff contain natural scene images. PascalVOC only contains 21 semantic classes (including background), making classification comparatively easy. Thus, the fraction of segment errors on PascalVOC is with 8.0% much smaller than for ADE20K (30.8%) and COCO-Stuff 164k (37.8%), having 150 and 171 classes, respectively. Also, object contours in PascalVOC are surrounded by a band of ignore pixels, leading to fewer boundary errors (3.6%) as well. On all three of these natural scene datasets, segment errors and extent errors dominate.

CityScapes is an autonomous driving dataset containing urban road scenes. It has high-quality annotations for 19 classes, enabling a high overall $mIoU$. With a segment er-

Setting	Architecture	mIoU	mE$_\star$oU			$\widetilde{\text{mE}_\star\text{oU}}$		
			bnd	*ext*	*seg*	*bnd*	*ext*	*seg*
Full sup.	DeeplabV3+ [9] (R101)	78.6	3.4	9.8	8.1	4.7	11.4	8.1
Weak sup. ($\mathcal{I}$)	BECO [27] (DeeplabV3+,R101)	70.8	4.1	18.0	7.1	6.1	20.1	7.1
Semi-sup. (1/16)	U2PL [31] (DeeplabV3+, R101)	68.0	4.6	14.6	12.8	8.1	18.0	12.8
Semi-sup. (1/8)	U2PL [31] (DeeplabV3+, R101)	71.4	3.9	12.1	12.6	5.8	15.9	12.6
Semi-sup. (1/4)	U2PL [31] (DeeplabV3+, R101)	74.8	4.4	11.3	9.5	6.1	13.4	9.5
Semi-sup. (1/2)	U2PL [31] (DeeplabV3+, R101)	78.4	3.7	9.9	8.0	5.0	11.6	8.0
Open vocabulary	ODISE [35] (Mask2Former, UNet)	83.9	1.5	6.5	8.1	2.0	7.3	8.1
Open vocabulary	TCL [4] (CLIP, ViT-B)	51.1	7.2	29.9	11.7	13.8	36.1	11.7

Table 4. **Results on PascalVOC 2012 val for different learning settings.** Number in parentheses for semi-sup. indicates the proportion of labeled samples (see [31]). The varying error rates for the different settings point out different challenges, *e.g.*, weak supervision leads to many extent errors, whereas open-vocabulary semantic segmentation produces relatively high segment error rates.

ror rate of 5.6%, classification on this dataset seems less challenging. The remote sensing dataset iSAID contains many small objects such as vehicles in overhead imagery, leading to a relatively high boundary error rate of 11.8%. A higher boundary error rate in our selection can only be observed on STARE, a medical image dataset for the segmentation of retinal vessels. As it only contains the classes "vessel" and "background", the segment error rate is only 0.2% on this dataset. Also, extent errors are very low, making boundary errors the main limiting factor. This is because the vessels in the dataset are usually a single large contiguous structure with very thin and fine elements.

Overall, we can see that, similar to model architectures, different datasets and different domains have rather distinctive features and come with different challenges, which is reflected in the error distributions. Therefore, we argue that our error analysis can help in selecting or developing a suitable segmentation architecture for a specific task.

4.5. Comparing Learning Settings

Apart from different model architectures and datasets, there are also various learning settings in semantic segmentation, for which we conduct a comparison in Table 4. We conduct the comparison on PascalVOC as it is still a highly popular benchmark for weakly and semi-supervised semantic segmentation. Looking at the image-level weakly supervised method BECO [27], we see that BECO is slightly superior to the fully supervised DeepLabV3+ in terms of segment errors (7.1% vs. 8.1%). However, the weak supervision does not provide any localization information, leading to higher numbers of boundary and extent errors. Extent errors are particularly high for BECO as segmenting non-discriminative parts is a key challenge in weakly supervised semantic segmentation [5, 21, 23, 24, 30].

For the semi-supervised method U2PL [31], we observe that the error rates decrease rather uniformly for all types as the number of supervised samples increases. Thus, we conclude that using more supervised samples for training does not resolve only specific error types, but it is beneficial for all of the three proposed error categories.

In addition to these weakly and semi-supervised approaches, we also assess two representatives of open-vocabulary semantic segmentation, a task that has attracted tremendous attention recently [4, 18, 26, 35, 36]. Although ODISE [35] achieves a strong $mIoU$ of 83.9%, its segment error rate of 8.1% is not lower than the one of the fully supervised DeepLabV3+ (8.1%) and the one of the weakly supervised BECO (7.1%). Also, TCL [4] produces many segment errors (11.7%). However, since TCL is supervised with only image-text pairs, it receives no localization information during training, leading to high boundary and extent errors. Altogether, the observed segment error rates indicate that, to this date, closed-vocabulary methods are stronger in terms of classification than open-vocabulary methods.

5. Conclusion

In this paper, we proposed an intuitive error categorization that allows the investigation of the strengths and weaknesses of semantic segmentation models in a quantitative way. We conducted an extensive analysis, including various semantic segmentation architectures, datasets, and learning settings. In doing so, we demonstrated the practical value of our approach and gained interesting insights. Most notably, high segment error rates revealed that mask-classification-based segmentation architectures such as Mask2Former and OneFormer have shortcomings in classification. A simple combination with other models producing fewer segment errors suffices to improve the performance and achieve new state-of-the-art results on ADE20K. This opens up a concrete direction on how to advance cutting-edge semantic segmentation models and underlines the usefulness of our proposed error analysis.

References

[1] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021. 6

[2] Daniel Bolya, Sean Foley, James Hays, and Judy Hoffman. Tide: A general toolbox for identifying object detection errors. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 558–573. Springer, 2020. 2

[3] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018. 5, 7

[4] Junbum Cha, Jonghwan Mun, and Byungseok Roh. Learning to generate text-grounded mask for open-world semantic segmentation from only image-text pairs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11165–11174, 2023. 8

[5] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation via sub-category exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8991–9000, 2020. 1, 8

[6] Liangyu Chen, Tong Yang, Xiangyu Zhang, Wei Zhang, and Jian Sun. Points as queries: Weakly semi-supervised object detection by points. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8823–8832, 2021. 2

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 2

[8] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 6

[9] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 6, 8

[10] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022. 2

[11] Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C Berg, and Alexander Kirillov. Boundary iou: Improving object-centric image segmentation evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15334–15342, 2021. 2, 5

[12] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. 2, 6

[13] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021. 2, 6

[14] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 5, 7

[15] Gabriela Csurka, Diane Larlus, Florent Perronnin, and France Meylan. What is a good evaluation measure for semantic segmentation?. In *Bmvc*, volume 27, pages 10–5244. Bristol, 2013. 2

[16] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. Deepglobe 2018: A challenge to parse the earth through satellite images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 172–181, 2018. 1

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 5, 7

[18] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European Conference on Computer Vision*, pages 540–557. Springer, 2022. 8

[19] AD Hoover, Valentina Kouznetsova, and Michael Goldbaum. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. *IEEE Transactions on Medical imaging*, 19(3):203–210, 2000. 5, 7

[20] Jitesh Jain, Jiachen Li, Mang Tik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. Oneformer: One transformer to rule universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2989–2998, 2023. 2, 6, 7

[21] Beomyoung Kim, Sangeun Han, and Junmo Kim. Discriminative region suppression for weakly-supervised semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1754–1761, 2021. 1, 8

[22] Pushmeet Kohli, L'ubor Ladickỳ, and Philip HS Torr. Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82:302–324, 2009. 2

[23] Jungbeom Lee, Jooyoung Choi, Jisoo Mok, and Sungroh Yoon. Reducing information bottleneck for weakly supervised semantic segmentation. *Advances in Neural Information Processing Systems*, 34:27408–27421, 2021. 1, 8

[24] Jing Li, Junsong Fan, and Zhaoxiang Zhang. Towards noiseless object contours for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16856–16865, 2022. 1, 8

[25] Yanghao Li, Saining Xie, Xinlei Chen, Piotr Dollar, Kaiming He, and Ross Girshick. Benchmarking detection

transfer learning with vision transformers. *arXiv preprint arXiv:2111.11429*, 2021. 2

[26] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7061–7070, 2023. 8

[27] Shenghai Rong, Bohai Tu, Zilei Wang, and Junjie Li. Boundary-enhanced co-training for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19574–19584, 2023. 8

[28] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7262–7272, 2021. 6

[29] Xian Sun, Peijin Wang, Zhiyuan Yan, Feng Xu, Ruiping Wang, Wenhui Diao, Jin Chen, Jihao Li, Yingchao Feng, Tao Xu, et al. Fair1m: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 184:116–130, 2022. 2

[30] Xiang Wang, Shaodi You, Xi Li, and Huimin Ma. Weakly-supervised semantic segmentation by iteratively mining common object features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1354–1362, 2018. 1, 8

[31] Yuchao Wang, Haochen Wang, Yujun Shen, Jingjing Fei, Wei Li, Guoqiang Jin, Liwei Wu, Rui Zhao, and Xinyi Le. Semi-supervised semantic segmentation using unreliable pseudo-labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4248–4257, 2022. 8

[32] Syed Waqas Zamir, Aditya Arora, Akshita Gupta, Salman Khan, Guolei Sun, Fahad Shahbaz Khan, Fan Zhu, Ling Shao, Gui-Song Xia, and Xiang Bai. isaid: A large-scale dataset for instance segmentation in aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 28–37, 2019. 5, 7

[33] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434, 2018. 6

[34] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34:12077–12090, 2021. 6

[35] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2955–2966, 2023. 8

[36] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *European Conference on Computer Vision*, pages 736–753. Springer, 2022. 8

[37] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 6

[38] Anlin Zheng, Yuang Zhang, Xiangyu Zhang, Xiaojuan Qi, and Jian Sun. Progressive end-to-end object detection in crowded scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 857–866, 2022. 2

[39] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, 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*, pages 6881–6890, 2021. 6

[40] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 5, 7

[41] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*, pages 3–11. Springer, 2018. 1