Intra-Batch Supervision for Panoptic Segmentation on High-Resolution Images

Daan de Geus   Gijs Dubbelman

Eindhoven University of Technology

d.c.d.geus@tue.nl, g.dubbelman@tue.nl

Abstract

Unified panoptic segmentation methods are achieving state-of-the-art results on several datasets. To achieve these results on high-resolution datasets, these methods apply crop-based training. In this work, we find that, although crop-based training is advantageous in general, it also has a harmful side-effect. Specifically, it limits the ability of unified networks to discriminate between large object instances, causing them to make predictions that are confused between multiple instances. To solve this, we propose Intra-Batch Supervision (IBS), which improves a network's ability to discriminate between instances by introducing additional supervision using multiple images from the same batch. We show that, with our IBS, we successfully address the confusion problem and consistently improve the performance of unified networks. For the high-resolution Cityscapes and Mapillary Vistas datasets, we achieve improvements of up to +2.5 on the Panoptic Quality for thing classes, and even more considerable gains of up to +5.8 on both the pixel accuracy and pixel precision, which we identify as better metrics to capture the confusion problem.

1. Introduction

Panoptic segmentation [12] is a scene understanding task that requires grouping all pixels of an image into segments with different semantic meaning, while also distinguishing between individual object instances. Where this task was first primarily tackled with multi-branch neural networks solving instance segmentation and semantic segmentation separately [11, 13, 15, 20, 28], unified image segmentation approaches are now gaining popularity [2, 3, 16, 26, 31]. These unified approaches aim to use the same unified architecture for all types of image segmentation, and they outperform the previous state-of-the-art on various datasets. Despite their good performance, unified methods still suffer from a *hitherto unidentified problem* when applied to high-resolution images, which is due to crop-based training (see Figure 1). In this work, we explore this problem, identify its cause, and we propose a method to solve it.
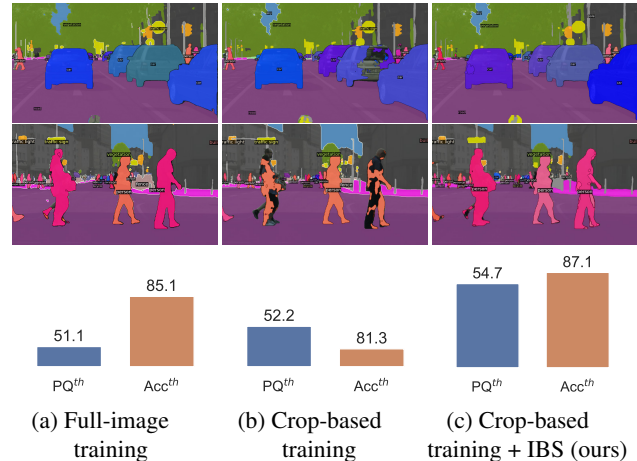

Figure 1: **Results for different training strategies**. Crop-based training of unified networks, which is needed to achieve a good segmentation performance on high-resolution images, leads to inaccurate predictions for large *thing* objects. We propose to solve this problem with Intra-Batch Supervision (IBS), and improve both the Panoptic Quality (PQ^{th}) [12] and pixel accuracy (Acc^{th}) for things.

Within panoptic segmentation, we distinguish between two types of classes: *things* and *stuff* [12]. Things are countable classes for which we can identify individual instances, such as car and person. For thing classes, panoptic segmentation requires segments for each unique instance. Stuff classes are uncountable, often amorphous regions, such as road and sky, for which panoptic segmentation requires just one segment per class. Recently, unified methods achieve state-of-the-art panoptic segmentation results by using a unified architecture to solve both stuff and thing segmentation. There are different variations to this approach, but the overall principle is the same: 1) it generates an *embedding* vector for each stuff and thing segment, 2) it generates a tensor with *features* for the entire image, and 3) it outputs a segmentation mask for each segment by taking the product of the embeddings and features.

Despite the improvements made by these unified approaches, they still suffer from a problem that — to the
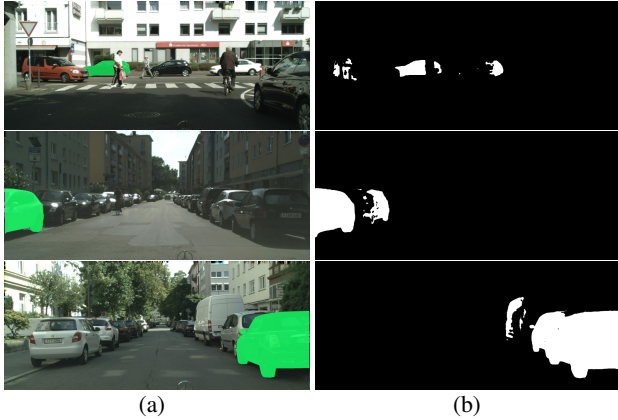
Figure 2: **Confusion between thing segments.** These are typical examples of the *confusion* in predicted thing segments for Panoptic FCN [16] when trained on image crops. (a) Input image with ground-truth thing mask in green. (b) The corresponding prediction confused between instances.

best of our knowledge – has not been reported in literature. Specifically, when trained and tested on high-resolution images, their predictions for individual thing segments often overlap multiple thing instances, especially for easily recognizable large objects, see Figure 2. In this work, we find out that this problem is caused by crop-based training, which is customary and often necessary for panoptic segmentation methods to be able to train on high-resolution images and achieve state-of-the-art results [23]. Specifically, unified networks generate thing segmentation masks by taking the product of the *embedding* for those segments, and the overall *features*. By supervising these masks during training, the network learns to generate embeddings and features that are sufficiently discriminative to generate an accurate, unique mask for each thing instance. However, when training on crops, the network sees considerably fewer instances in an image than during testing, as illustrated in Figure 3. Because the network is trained to distinguish only a relatively small number of objects, it is not sufficiently discriminative for the large number of objects encountered during testing. This causes the problem visualized in Figure 2, where segmentation masks are confused between multiple thing instances. We further explore this confusion problem in Section 3. We show that the standard Panoptic Quality metric [12] for panoptic segmentation does not adequately reflect the confusion problem because it is biased towards small segments and infrequent classes. We find that the pixel accuracy and pixel precision metrics better reflect the confusion problem, and we propose to use these metrics together with the PQ.

To solve the confusion problem caused by crop-based training, the network should receive more adequate supervision for thing segments. To be generally applicable, this should be achieved without using additional data or chang-
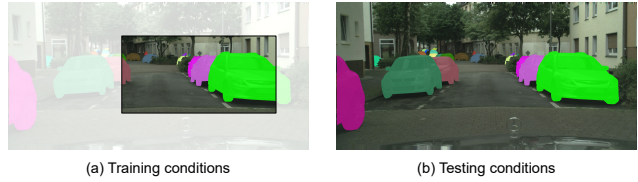

Figure 3: **Crop-based training and full-image inference**. (a) When training with image crops, the network can only learn to discriminate between instances within a small part of the input image. (b) During testing, which happens on full images, the network needs to discriminate between many more instances. As it has not learnt to do so, this will lead to confusion between instances (see Figure 2).

ing the network architecture. In this paper, we propose to achieve this by teaching the network to distinguish instances not only from other instances in the same image crop, but also from instances in other image crops in the same training batch. Specifically, we take the product of the thing embeddings from one crop and the features from another crop, and supervise the outputs to teach the network to be more discriminative. We call our improvement *Intra-Batch Supervision* (IBS), as we apply additional supervision to the product of thing embeddings and features of different image crops in the same batch. Because IBS only helps if there is meaningful supervision, we also introduce a new crop sampling strategy that makes sure that IBS always occurs between crops for which the confusion problem is likely to occur. In Section 4, we motivate and describe IBS in detail.

In Section 6, we extensively evaluate our proposed solution on multiple networks and datasets, using the experimental setup as described in Section 5. We find that our IBS significantly decreases the confusion between thing segments, as intended, resulting in a consistent increase of the PQ for thing classes, and a significant improvement on the pixel accuracy. This is also shown in Figure 1. Further discussions and conclusions are provided in Sections 7 and 8. All code is available through `https://ddegeus.github.io/intra-batch-supervision/`.

In summary, we make the following contributions:

- We demonstrate that crop-based training of unified segmentation networks has a harmful side-effect that leads to predictions with *confusion* between thing segments, and we show that standard metrics for panoptic segmentation do not capture this problem.
- We propose *intra-batch supervision* (IBS) to solve the confusion problem, which introduces additional supervision to allow the network to better discriminate between thing instances.
- We show the effectiveness of IBS on state-of-the-art unified panoptic segmentation and instance segmentation networks, on two different datasets.

2. Related work

Multi-branch panoptic segmentation. Following the formalization of the panoptic segmentation task [12], many deep neural network architectures have been proposed to solve it. As panoptic segmentation is essentially a combination of semantic segmentation and instance segmentation, most proposed methods initially used neural networks that make separate predictions for these two tasks [5, 11, 15, 19, 20, 22, 28]. Concretely, this means that there is a joint backbone which extracts features from the image, followed by two separate branches: 1) one that predicts instance masks and classes for *thing* classes, often based on Mask R-CNN [8], and 2) another that applies per-pixel classification to segment the remaining *stuff* classes. Over the past years, many variations and improvements to this multi-branch framework have been introduced [1, 6, 7, 14, 24, 29, 30], but all these networks use separate, task-specific architectural components for instance and semantic segmentation.

Unified panoptic segmentation. Recently, there has been an effort to develop universal or *unified* approaches for the panoptic segmentation task. The proposed unified networks [2, 3, 16, 26, 31] treat all segments, *i.e.*, stuff and things, as equally as possible. Although there are clear differences between the unified methods (*e.g.*, fully-convolutional [16] vs. transformer-based [2, 3]), the overall principle is similar: images are segmented by taking the product of per-segment *embeddings* and the overall pixel-level *features* from the images. This principle is similar to an approach used by recent instance segmentation methods [25, 27], but they do not apply stuff segmentation. By following this approach, the panoptic networks do not need to use multiple, specialized branches, and achieve state-of-the-art results. In this work, we show that despite the state-of-the-art performance, these models still make specific inaccurate predictions when applied on high-resolution images, and we propose a method to solve it.

Crop-based training. Many panoptic segmentation networks use crop-based training when applied to high-resolution datasets [1, 2, 11, 16, 20], and Porzi *et al*. [23] show that crop-based training on high-resolution images outperforms full-image training, *e.g.*, because it allows for more stable training and facilitates identifying small objects. At the same time, Porzi *et al*. also highlight that crop-based training has negative side-effects for multi-branch panoptic segmentation methods that apply box-based detection. They show that 1) bounding boxes are truncated due to cropping, and 2) large images are underrepresented in cropped images, which harms the segmentation performance. To solve this, they propose a new loss and a new image sampling strategy, respectively. In this work, we find that crop-based training is also advantageous for unified panoptic segmentation networks, but that it has a specific negative side-effect, which does not apply for multi-branch methods, and is not solved by the solutions proposed by Porzi *et al*. [23]. Therefore, this problem requires a new solution, which we propose and call *intra-batch supervision*.

3. Confusion problem analysis

When testing unified panoptic segmentation networks on high-resolution images, we observe a large number of inaccurately predicted segmentation masks for things. Specifically, these networks predict masks that overlap *multiple* thing instances, usually from the same semantic class, as if they are *confused*. Examples of individual confused thing predictions are shown in Figure 2, and the effect on the final panoptic segmentation result is visualized in Figure 6. We hypothesize that this problem is caused by the way these networks are trained, *i.e.*, with image crops. In this section, we explore the *confusion* problem in more detail.

3.1. Unified panoptic segmentation

Unified panoptic segmentation methods [2, 3, 16, 26, 31] treat thing and stuff segments as equally as possible. This means that the output masks for all thing and stuff segments are generated in a unified way, using the same operations.

The existing unified panoptic segmentation methods use the same high-level principle. In one part of the network, they generate *embeddings* $E \in \mathbb{R}^{N \times C}$ for all N *thing* and *stuff* segments in the input image, where C is the embedding dimension. These embeddings should encode the properties of the segment they are representing. In another part of the network, they generate a single tensor of *features* $F \in \mathbb{R}^{C \times H \times W}$, where H, W are the height and width of the image. These features should represent the content of each pixel as well as possible. Together, these per-segment embeddings and the per-image, pixel-level features are used to generate the panoptic segmentation output. Specifically, the output segmentation masks $M \in \mathbb{R}^{N \times H \times W}$ are the result of a matrix multiplication between embeddings E and features F, followed by a sigmoid activation. This process is visualized for thing segments in Figure 4 (top). To get the output in the panoptic segmentation format, each predicted mask should also have a corresponding class prediction. Each unified method treats this problem differently, *e.g.*, MaskFormer [3] applies a linear layer to the embeddings to predict the classes, and Panoptic FCN [16] already predicts a class for each embedding earlier in the network.

3.2. Effect of crop-based training

Panoptic segmentation requires that there is a *unique* segmentation mask for each thing instance in an input image. In unified networks, the output mask for each thing instance is the product of the per-segment *embedding* E

Network	Crop	PQ↑	PQ$^{\text{th}}$↑	PQ$^{\text{st}}$↑	Acc$^{\text{th}}$↑	Prec$^{\text{th}}$↑
		Cityscapes *val*				
Panoptic FCN [16]	✗	57.2	51.1	61.7	**85.1**	**91.4**
Panoptic FCN [16]	✓	59.5	**52.2**	**64.8**	81.3	86.8
		Mapillary Vistas *validation*				
Panoptic FCN [16]	✗	32.7	29.8	36.6	73.8	**80.8**
Panoptic FCN [16]	✓	**35.5**	**31.8**	**40.3**	**74.7**	77.9

Table 1: **Full-image vs. crop-based training.**

and per-image *features F*. Therefore, to get a unique mask for each instance, it is important that both of these components are sufficiently discriminative, *i.e.*, each per-segment embedding should be clearly distinguishable from the other generated embeddings, and the features at the pixels where an instance is present should be sufficiently different from the features at pixels with other instances. If this were not the case, the matrix multiplication between E and F would result in segmentation masks that have high activations at the pixels of more than one instance, *i.e.*, *confusion* between instances.

When training on full-resolution images, the network is able to learn embeddings and features that are sufficiently discriminative, as it sees many instances of different sizes and classes at different locations, which are representative of the situations it could encounter during testing. When training on image crops, the situation is different. Specifically, the 'field of view' of the network is limited. Instead of seeing full-resolution images, representative of those that it will encounter during testing, it will see only a fraction of the input images (see Figure 3). Consequently, the network sees significantly fewer instances during training than during testing. This hinders the network's ability to discriminate between many different instances, as it only learns to be discriminative among the small set of instances in a crop. We expect that this lack of sufficient 'negative' supervision causes the network to output *thing* segmentation masks with confusion between different instances, when applied to full-resolution images during testing. Because only few large segments fit in a crop, the network sees few large segments during training, so we expect that this problem mainly affects these large segments.

3.3. Impact on metrics

To verify our hypothesis that crop-based training causes the *confusion* problem, and to measure the effect of the problem on the overall panoptic segmentation results, we train unified networks on (a) full images and (b) image crops, and report the performance in Table 1. As a first metric, we use the Panoptic Quality (PQ) [12], the standard metric for panoptic segmentation, defined as

$$\text{PQ} = \frac{1}{|C|} \sum_{c=1}^{|C|} \frac{\sum_{(p,g) \in TP_c} \text{IoU}(p,g)}{|TP_c| + \frac{1}{2}|FP_c| + \frac{1}{2}|FN_c|}, \quad (1)$$

where $|C|$ is the set of all classes, p and g are the predictions and ground-truth segments, and TP_c, FP_c and FN_c are the sets of true positives, false positives and false negatives per class c, respectively. True positives are matched prediction/ground-truth pairs that have an intersection-over-union (IoU) larger than 0.5. False positives are predictions not matched to a ground-truth segment, and false negatives are ground-truth segments not matched to a prediction. By design, this metric treats all classes equally, as it takes the average over all classes, and it treats all segments equally, as they all count as a single TP, FP or FP.

From the results in Table 1, we observe that the PQ increases when we use crop-based training instead of full-image training, for both thing (PQ$^{\text{th}}$) and stuff classes (PQ$^{\text{st}}$). This is as expected, because crop-based training facilitates more stable training and allows for better identification of small and underrepresented segments [23], thereby improving the quality of the predictions. However, from the qualitative results in Figures 1 and 2, it is also obvious that there are many inaccurate predictions for large objects when using crop-based training, which can be harmful for downstream tasks. Because this problem predominantly occurs for large segments and classes with multiple segments per image, and the PQ treats all segments and classes equally, the segments affected by the confusion problem have only a limited impact on the PQ, compared to the many small segments and the segments from other classes. Therefore, we also report the performance on the previously commonly-used pixel accuracy metric [10], and the related pixel precision. Because we are mainly interested in the thing classes, we report on these metrics for things only, with Acc$^{\text{th}}$ and Prec$^{\text{th}}$, respectively. They are defined as

$$\text{Acc}^{\text{th}} = \frac{|TP_{th}^{px}|}{N_{th,g}^{px}}; \quad (2)$$

$$\text{Prec}^{\text{th}} = \frac{|TP_{th}^{px}|}{|TP_{th}^{px}| + |FP_{th}^{px}|}, \quad (3)$$

where TP_{th}^{px} and FP_{th}^{px} are the sets of true positive and false positive thing pixels, respectively, and $N_{th,g}^{px}$ is the total number of ground-truth thing pixels. We count a pixel as a true positive if 1) it belongs to a predicted thing segment that has an IoU > 0.5 with a ground-truth segment of the same class (like for the PQ), and 2) if this pixel intersects with the matched ground-truth segment. In other words, the Acc$^{\text{th}}$ captures how many ground-truth thing pixels have been correctly predicted, and the Prec$^{\text{th}}$ captures how many thing predictions are correct. Ideally, panoptic segmentation methods should score well on *all three* metrics, the PQ, the Acc$^{\text{th}}$ and the Prec$^{\text{th}}$, as this means that A) they work well on average, over all classes and segment sizes, and B) they make correct predictions for most pixels in an image.

When looking at the results for crop-based training in Table 1 in terms of the Acc$^{\text{th}}$, we observe a drop or much more
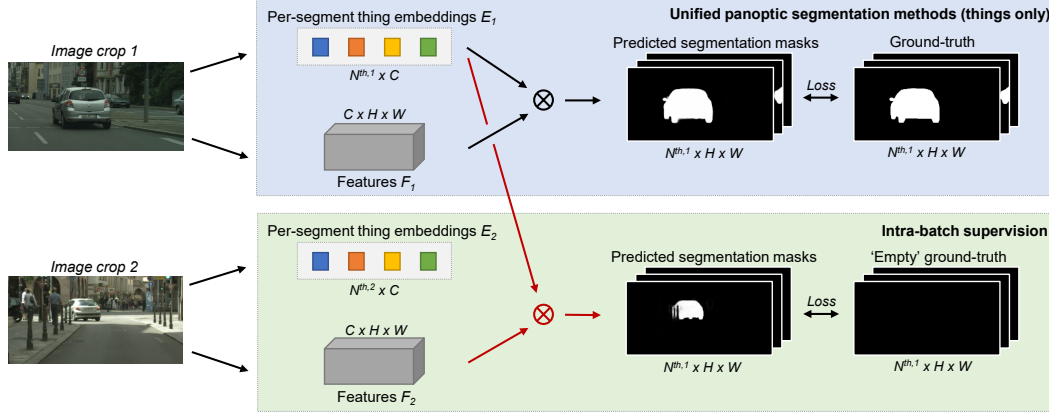
Figure 4: **Intra-Batch Supervision**. We take the product of thing embeddings E from one image crop and features F from another image crop in the batch (indicated in red), and as the thing instances belonging to the embeddings are not present in the other image crop, the result should be 'empty' predicted masks. By applying a loss to these predictions, we teach the network to generate more discriminative embeddings and features, leading to predicted masks with less *confusion*.

modest increase than for the PQ, and we observe a consistent, significant drop for the Prec[th]. This shows that, as expected and as observed in Figure 2, crop-based training has a harmful side-effect that causes a large number of pixels to be predicted inaccurately. This side-effect can clearly cause severe problems for downstream tasks such as autonomous driving because of its impact on the identification of large, nearby objects, and thus it needs to be solved.

4. Intra-Batch Supervision

In Section 3, we analyzed the harmful side-effect of crop-based training that causes *confused predictions*, and hypothesized that it is caused by a lack of sufficient supervision during training. We aim to solve this problem by proposing a method that is easily and generally applicable, so it should not require a) changes to the network architecture, or b) the use of any additional training data. To achieve this, we propose to address this problem for unified networks by introducing additional supervision, using the *embeddings* and *features* that are already being generated within each training batch. We call our proposed solution *Intra-Batch Supervision* (IBS). Because our method is only applied during training, it does not add any computational complexity during testing.

4.1. Additional supervision

The principle of IBS is straightforward: take the product of the per-segment *embeddings* E of one image crop and the per-image-crop *features* F of another crop in the batch, and apply a loss to the resulting segmentation masks. Under the assumption that each object instance only occurs in one crop, this means that for embeddings E belonging to thing segments, matrix multiplication with features F from another crop should always result in empty segmentation

masks, as the instances represented by E are not present in the other crop. When this is supervised, the network will learn to generate embeddings and features that are more unique to the segments they correspond to. By applying IBS to random *embedding-feature* combinations in the batch, the network receives supervision for a large number of instances which are outside its 'field-of-view', solving the supervision problem caused by crop-based training. The key principle of IBS is visualized in Figure 4. Note that we expect that the assumption on each thing instance occurring only in a single crop per batch is violated so infrequently, due to the large dataset size and the random batch and crop sampling methods, that its effect is negligible.

Predictions. The first step of intra-batch supervision consists of making the intra-batch predictions, between the *embeddings* and *features* of different image crops. Consider a training scenario where the batch size is B. With a given unified method, we generate embeddings $E_1, ..., E_B$ and features $F_1, ..., F_B$ for B image crops. Then, for each *source* crop, *i.e.*, the crop from which we take the embeddings, we randomly pick another crop from the batch to be the *target* crop, from which we pick the features. For the embeddings E from each source crop i we then have corresponding features F from target crop tgt_i, resulting in E,F pairs $\{E_1, F_{tgt_1}\}, ..., \{E_B, F_{tgt_B}\}$. Subsequently, each of these pairs are subjected to matrix multiplication and sigmoid activation to generate the segmentation masks of segments from source crop i in target crop tgt_i, *i.e.*, $M_{i \rightarrow tgt_i}$.

For example, when $B = 3$, we could randomly pick pairs $\{E_1, F_2\}$, $\{E_2, F_3\}$, $\{E_3, F_1\}$. We then apply a matrix multiplication between E_1 and F_2, to generate the segmentation masks for embeddings from crop 1 on crop 2, $M_{1 \rightarrow 2}$. The example of pair $\{E_1, F_2\}$ is visualized in Figure 4.

Supervision. The activations in segmentation mask $M_{i\to tgt_i}$ represent the presence of segments from source crop i in target crop tgt_i. Under the assumption that each instance is present in only one crop, this means that the predictions for thing segment embeddings should always be zero. If it is not zero, this means that the network confuses a thing instance from i with another instance in tgt_i, which is the *confusion* we are trying to solve. Therefore, we supervise these segmentation masks in such a way that the network learns to output *empty* segmentation masks for all things in $M_{i\to tgt_i}$, creating unique embeddings and features for each instance. As we only want to apply the loss to the masks that indicate thing instances, we identify and retrieve the masks generated by embeddings E for which there is an associated *thing* ground-truth segment. For the segmentation masks of the $N^{th,i}$ things in the source crop i, $M_{i\to tgt_i}^{th} \in \mathbb{R}^{N^{th,i}\times H\times W}$, we generate an artificial segmentation ground-truth S^{th}, which has the shape of $M_{i\to tgt_i}^{th}$, but is zero at all pixels. We supervise this by applying the focal loss [17] to the generated ground-truth and mask logits, to force the predictions to zero. We choose the focal loss because it is designed for data with large class imbalances, and we expect the majority of the predictions to be zero.

4.2. Crop sampling

For intra-batch supervision to work, it is critical that the supervision that the network receives is meaningful. When solving the confusion problem, supervision is meaningful when the *intra-batch predictions* $M_{i\to tgt_i}$ contain positive, 'confused' masks. When this is the case, the embeddings from image A activate features in image B, meaning that they are not sufficiently discriminative, and we can apply a loss to counter this. As we observe that confusion mainly happens between thing instances of the same class, we expect that meaningful supervision mostly occurs when both crops contain objects from the same class. When images are sampled and cropped randomly, it is unlikely that this is always the case, especially when the images are very large and when there are many classes in the dataset.

Therefore, we propose a new crop sampling strategy, loosely inspired by the Instance Scale-Uniform Sampling (ISUS) strategy presented by Porzi *et al*. [23]. Where they use a uniform sampling strategy to make sure that the network sees segments of all classes and sizes at a somewhat consistent rate, we want to sample the crops in such a way that there are always two crops that contain a segment of the same class and a similar size. Specifically, we 1) sample a class, 2) pick two images that contain a segment of this class, 3) pick one random ground-truth segment from that class from each image, 4) resize the images so that these segments have a similar size, and 5) crop the images in such a way that the crops contain the sampled ground-truth segments. We then make sure that IBS is applied between the

crops that contain different segments of the same class. This way, we aim to accomplish meaningful supervision as often as possible, enhancing the effect of IBS.

5. Experimental setup

We conduct several experiments to show the effectiveness of our IBS approach to solve the confusion problem. Concretely, we use state-of-the-art, unified panoptic segmentation methods Panoptic FCN [16] and Mask2Former [2], and evaluate them with and without IBS, on Cityscapes and Mapillary Vistas, which are *de facto* benchmarks for high-resolution panoptic segmentation. We use the evaluation metrics described in Section 3.3.

5.1. Datasets

We train and evaluate on two datasets with high-resolution street scene images: Cityscapes [4] and Mapillary Vistas [21]. The Cityscapes dataset contains 5k images, captured in and around Germany. All images have a resolution of 1024×2048 pixels, and there are panoptic labels for 19 classes (8 *thing*, 11 *stuff*). We train on the *train* set (2975 images) and evaluate on the *val* set (500 images). Mapillary Vistas [21] is much larger: it has 25k images of different, large resolutions (8.9×10^6 pixels on average), and has labels for 65 classes (37 *thing*, 28 *stuff*). We train on the *training* set (18k images) and evaluate on the *validation* set (2k images).

5.2. Implementation details

All networks are trained on Nvidia A100 GPUs with 40GB memory each. For Cityscapes, we use 4 GPUs for training; for Mapillary Vistas we use 8. All networks use a ResNet-50 backbone [9], for a fair comparison. For both Panoptic FCN [16] and Mask2Former [2], we extend their official published code and training settings, and crop-based training is conducted exactly like in the original works. Below, we specify the most relevant hyperparameters and settings. For more details, we refer to the original work [2, 16] and our supplementary material.

Panoptic FCN. For crop-based training on Cityscapes, we train for 65k steps with batches of 32 crops of 512×1024 pixels, after randomly resizing the image with a factor between 0.5 and 2.0. For full-image training, we train the network for 100k steps on batches of 4 images, after randomly resizing them with a factor between 0.5 and 2.0. For Mapillary Vistas, we train for 150k steps with 32 crops of 1024×1024 pixels, taken from images which are first randomly resized such that the shortest side is between 1024 and 2048 pixels. For full-image training, we train with batches of 8 images, for 150k steps, after randomly resizing the shortest side between 1024 and 2048 pixels.

Mask2Former. For Mask2Former, we make an important improvement to the code which is relevant for crop-

Network	PQ↑	PQth↑	PQst↑	Accth↑	Precth↑
		Cityscapes *val*			
Panoptic FCN [16]	59.5	52.2	64.8	81.3	86.8
Panoptic FCN [16] + IBS (ours)	**60.8**	**54.7**	**65.3**	**87.1**	**92.6**
Mask2Former [2]	62.1	55.2	67.2	87.1	93.3
Mask2Former [2] + IBS (ours)	**62.4**	**55.7**	**67.3**	**87.6**	**94.1**
		Mapillary Vistas *validation*			
Panoptic FCN [16]	35.5	31.8	**40.3**	74.7	77.9
Panoptic FCN [16] + IBS (ours)	**36.3**	**33.6**	40.0	**77.0**	**82.2**
Mask2Former [2]	41.5	33.3	**52.4**	71.7	78.8
Mask2Former [2]+ IBS (ours)	**42.2**	**34.9**	52.0	**75.7**	**84.1**

Table 2: **Main results.** We apply our IBS to Panoptic FCN [16] and Mask2Former [2], and train and evaluate on the Cityscapes and Mapillary Vistas datasets [4, 21].

based training, *i.e.*, removing ground-truth segments outside the cropped region. Due to bipartite matching, these regions required a prediction, which the network could not make. By removing these segments during training, the baseline Panoptic Quality for Mapillary Vistas is significantly improved (36.3 [2] vs. 41.5 in Table 2). For Cityscapes, we train for 90k steps on batches of 16 crops of 512×1024 pixels. For Mapillary Vistas, we train for 300k steps on batches of 16 crops of 1024×1024 pixels.

6. Results

6.1. Intra-Batch Supervision

We apply our Intra-Batch Supervision (IBS) to two state-of-the-art unified panoptic segmentation methods, and evaluate the performance on two high-resolution datasets in Table 2. From these results, we find that IBS yields a consistent but modest improvement on the Panoptic Quality (PQ) metric [12]. This is mainly caused by a boost in the PQ for things classes, PQth, which consistently shows considerable improvements. This is expected, because the aim of our IBS is to improve the confusion problem that occurs for thing segments. As explained in Section 3.3, however, the ability of the PQ metric to capture errors for large objects of frequently-occuring classes, such as the *confusion* problem that we are addressing, is limited. To better capture this, we also report on the pixel accuracy (Accth) and pixel precision (Precth) for things classes. For these metrics, we observe much more significant improvements, *e.g.*, a +5.8 improvement for both metrics for Panoptic FCN on Cityscapes, and +4.0 and +5.3 for Mask2Former on Mapillary Vistas. In other words, IBS causes the network to correctly predict the panoptic segment for significantly more pixels than methods that used the naive crop-based training. This is also clearly visible in the qualitative examples shown in Figure 6 and the supplementary material. Where naive crop-based training causes inaccurate thing segmentation masks, IBS solves this problem.

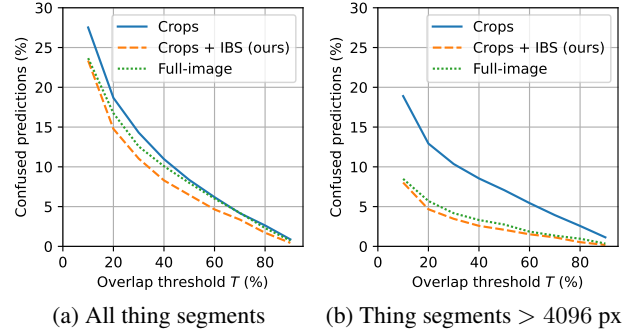(a) All thing segments   (b) Thing segments > 4096 px

Figure 5: **Confusion.** Ratio of predicted thing segments that intersect with at least *two* ground-truth thing segments for more than $T\%$ of the ground-truth area, on Cityscapes *val* [4], for different training settings. Lower is better.

Crop	IBS	Sampling	PQ↑	PQth↑	PQst↑	Accth↑	Precth↑
✗	✗	random	32.7	29.8	36.6	73.8	80.8
✓	✗	random	35.5	31.8	**40.3**	74.7	77.9
✓	✓	random	35.9	32.9	39.9	76.3	**82.5**
✓	✓	ours	**36.3**	**33.6**	40.0	**77.0**	82.2

Table 3: **Ablations.** Impact of different IBS components on Panoptic FCN [16] on Mapillary Vistas *validation* [21].

6.2. Confusion

To assess the effect of IBS on the specific confusion problem that we identified, instead of the more general metrics, we conduct an additional experiment where we calculate how often confusion occurs for predicted thing segments. Concretely, for different thresholds T, we calculate how many predicted thing segments intersect with *at least 2 ground-truth thing segments* for more than $T\%$ of the ground-truth area. The resulting number will indicate *how many predicted thing segments are confused between at least two segments* given an overlap threshold T. In Figure 5, we show the results for this measure for full-image training, naive crop-based training and crop-based training with IBS. We find that, as we hypothesized, naive crop-based training leads to considerably more confusion than full-image training, especially for large thing segments. When we introduce our IBS, however, this confusion drops significantly. These results confirm that the confusion problem caused by crop-based training mainly impacts large segments, and that our IBS effectively solves this problem.

6.3. Ablations

In Table 3, we show the effect of the different components of our proposed IBS. From these results, we find that vanilla IBS with random crop sampling already leads to a considerable improvement on the pixel accuracy and pixel precision for things, and a moderate improvement on the PQ. When we introduce our new sampling strategy, which

Figure 6: **Results for crop-based training on Panoptic FCN.** Top: without IBS, bottom: with IBS. From left to right, first two images: Cityscapes *val* set [4]; final two: Mapillary Vistas *validation* set [21]. Each segment gets a unique color and text label. The most prominent examples of predictions with *confusion* are indicated with red rectangles. Best viewed digitally.

makes sure that IBS is applied to images containing segments from the same semantic class, both the pixel accuracy and the PQ increase even further. This shows that IBS is effective, and that an effective crop sampling strategy, which aims to boost the amount of *meaningful* supervision, is required to unlock its full potential.

6.4. Instance segmentation

Because unified networks can also be used for instance segmentation, we expect the confusion problem to occur here as well. To show the more general applicability of our IBS to other computer visions tasks, we apply IBS to Mask2Former [2] for instance segmentation on Mapillary Vistas [21]. In Table 4, we report the results on the Average Prediction (AP), the standard COCO metric [18]. We find that our IBS leads to an improvement on the AP, especially for the large instances (AP^L) that are most impacted by confusion. Like for panoptic segmentation, we also find significant boosts for the Acc^{th} and $Prec^{th}$, with +4.3 for both metrics. This shows that IBS can also effectively solve the confusion problem for instance segmentation.

7. Discussion

With experiments, we found that the confusion problem is not effectively captured by the Panoptic Quality (PQ) metric, because large thing segments have only limited impact. Instead, we found that other established segmentation metrics, *i.e.*, the pixel accuracy and pixel precision, are better suited to measure the confusion problem. Therefore, although the PQ is an essential metric – as it is important that a method works well on segments of all categories and different sizes – we advocate for the use of *multiple* metrics to evaluate methods, and not just a single metric like the PQ, to properly assess their performance for a variety of downstream tasks. We hypothesize that focusing only on a single metric is one of the reasons that the confusion problem has not been addressed until now, despite its relevance for

Network	AP↑	AP^S↑	AP^M↑	AP^L↑	Acc^{th}↑	$Prec^{th}$↑
Mask2Former [2]	17.4	**5.3**	18.2	34.9	73.4	74.9
Mask2Former [2] + IBS (ours)	**18.1**	4.8	**18.5**	**36.8**	**77.7**	**79.2**

Table 4: **Instance segmentation.** IBS results for instance segmentation with Mask2Former [2] evaluated on the Mapillary Vistas *validation* set [21].

downstream tasks such as autonomous driving and robotics because of its impact on the performance for large objects.

8. Conclusions

In this work, we have shown that the generally advantageous crop-based training strategy has a negative side-effect that causes state-of-the-art, unified panoptic segmentation networks such as Panoptic FCN [16] and Mask2Former [2] to predict inaccurate segmentation masks for thing instances. To solve this, we have proposed Intra-Batch Supervision (IBS), which introduces additional supervision for thing segments by teaching the network to distinguish thing instances from instances in other crops. With experiments, we have shown that our IBS solves the confusion problem for multiple unified panoptic segmentation networks on two high-resolution datasets, causing consistent improvements on the Panoptic Quality for things, and large gains on the pixel accuracy and pixel prediction metrics. Moreover, we have demonstrated that IBS is effective on the instance segmentation task as well. These results demonstrate that our IBS is necessary to make accurate predictions on high-resolution datasets with state-of-the-art, unified networks.

References

[1] Bowen Cheng, Maxwell D. Collins, Yukun Zhu, Ting Liu, Thomas S. Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-DeepLab: A Simple, Strong, and Fast Baseline for Bottom-Up Panoptic Segmentation. In *CVPR*, 2020.

[2] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention Mask Transformer for Universal Image Segmentation. In *CVPR*, 2022.

[3] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-Pixel Classification is Not All You Need for Semantic Segmentation. In *NeurIPS*, 2021.

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *CVPR*, 2016.

[5] Daan de Geus, Panagiotis Meletis, and Gijs Dubbelman. Panoptic Segmentation with a Joint Semantic and Instance Segmentation Network. *arXiv preprint arXiv:1809.02110*, 2018.

[6] Daan de Geus, Panagiotis Meletis, and Gijs Dubbelman. Fast Panoptic Segmentation Network. *IEEE Robotics and Automation Letters*, 5(2):1742–1749, 2020.

[7] Naiyu Gao, Yanhu Shan, Yupei Wang, Xin Zhao, Yinan Yu, Ming Yang, and Kaiqi Huang. SSAP: Single-Shot Instance Segmentation With Affinity Pyramid. In *ICCV*, 2019.

[8] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016.

[10] Juana Valeria Hurtado and Abhinav Valada. Semantic Scene Segmentation for Robotics. In Alexandros Iosifidis and Anastasios Tefas, editors, *Deep Learning for Robot Perception and Cognition*, pages 279–311. Academic Press, 2022.

[11] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollar. Panoptic feature pyramid networks. In *CVPR*, 2019.

[12] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollar. Panoptic Segmentation. In *CVPR*, 2019.

[13] Jie Li, Allan Raventos, Arjun Bhargava, Takaaki Tagawa, and Adrien Gaidon. Learning to Fuse Things and Stuff. *arXiv preprint arXiv:1812.01192*, 2018.

[14] Qizhu Li, Xiaojuan Qi, and Philip H.S. Torr. Unifying Training and Inference for Panoptic Segmentation. In *CVPR*, 2020.

[15] Yanwei Li, Xinze Chen, Zheng Zhu, Lingxi Xie, Guan Huang, Dalong Du, and Xingang Wang. Attention-Guided Unified Network for Panoptic Segmentation. In *CVPR*, 2019.

[16] Yanwei Li, Hengshuang Zhao, Xiaojuan Qi, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Fully Convolutional Networks for Panoptic Segmentation. In *CVPR*, 2021.

[17] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal Loss for Dense Object Detection. In *ICCV*, 2017.

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *ECCV*, 2014.

[19] Huanyu Liu, Chao Peng, Changqian Yu, Jingbo Wang, Xu Liu, Gang Yu, and Wei Jiang. An End-to-End Network for Panoptic Segmentation. In *CVPR*, 2019.

[20] Rohit Mohan and Abhinav Valada. EfficientPS: Efficient Panoptic Segmentation. *IJCV*, 2021.

[21] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes. In *ICCV*, 2017.

[22] Lorenzo Porzi, Samuel Rota Bulo, Aleksander Colovic, and Peter Kontschieder. Seamless Scene Segmentation. In *CVPR*, 2019.

[23] Lorenzo Porzi, Samuel Rota Bulo, and Peter Kontschieder. Improving Panoptic Segmentation at All Scales. In *CVPR*, 2021.

[24] Konstantin Sofiiuk, Olga Barinova, and Anton Konushin. AdaptIS: Adaptive Instance Selection Network. In *ICCV*, 2019.

[25] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional Convolutions for Instance Segmentation. In *ECCV*, 2020.

[26] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. MaX-DeepLab: End-to-End Panoptic Segmentation With Mask Transformers. In *CVPR*, 2021.

[27] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. SOLOv2: Dynamic and Fast Instance Segmentation. *NeurIPS*, 2020.

[28] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. UPSNet: A Unified Panoptic Segmentation Network. In *CVPR*, 2019.

[29] Tien-Ju Yang, Maxwell D Collins, Yukun Zhu, Jyh-Jing Hwang, Ting Liu, Xiao Zhang, Vivienne Sze, George Papandreou, and Liang-Chieh Chen. DeeperLab: Single-Shot Image Parser. *arXiv:1902.05093*, 2019.

[30] Yibo Yang, Hongyang Li, Xia Li, Qijie Zhao, Jianlong Wu, and Zhouchen Lin. SOGNet: Scene Overlap Graph Network for Panoptic Segmentation. In *AAAI*, 2020.

[31] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-Net: Towards Unified Image Segmentation. In *NeurIPS*, 2021.