Multi-task Fusion for Efficient Panoptic-Part Segmentation

Sravan Kumar Jagadeesh, René Schuster, and Didier Stricker
DFKI – German Research Center for Artificial Intelligence, Kaiserslautern, Germany
firstname.lastname@dfki.de

Keywords: Semantic segmentation, Instance segmentation, Panoptic segmentation, Part segmentation, Part-aware panoptic segmentation.

Abstract: In this paper, we introduce a novel network that generates semantic, instance, and part segmentation using a shared encoder and effectively fuses them to achieve panoptic-part segmentation. Unifying these three segmentation problems allows for mutually improved and consistent representation learning. To fuse the predictions of all three heads efficiently, we introduce a parameter-free joint fusion module that dynamically balances the logits and fuses them to create panoptic-part segmentation. Our method is evaluated on the Cityscapes Panoptic Parts (CPP) and Pascal Panoptic Parts (PPP) datasets. For CPP, the *PartPQ* of our proposed model with joint fusion surpasses the previous state-of-the-art by 1.6 and 4.7 percentage points for all areas and segments with parts, respectively. On PPP, our joint fusion outperforms a model using the previous top-down merging strategy by 3.3 percentage points in *PartPQ* and 10.5 percentage points in *PartPQ* for partitionable classes.

1 INTRODUCTION

The human eye can observe a scene at various levels of abstraction. Humans can not only view the scene and differentiate semantic categories such as bus, car, and sky, but they can also understand them. However, they can also distinguish between the parts of each entity, such as car windows and bus chassis, and group them according to their instances. There is no deep learning approach that seeks to achieve several layers of abstraction with a single network at the moment.

The two pieces that make up a scene are *stuff* and *things* (Cordts et al., 2016). Things are countable amorphous objects such as persons, cars, or buses, whereas *stuff* like the sky or road is usually not countable. Many tasks have been created to identify these aspects in an image. Semantic segmentation and instance segmentation are two of the most common tasks.

However, these methods are incapable of describing the entire image. Scene parsing was created to fill this void, with the goal of describing the entire image by recognizing and semantically segmenting both *stuff* and *things*, a process which is known as panoptic segmentation (Kirillov et al., 2019b). This approach has introduced several state-of-the-art panoptic segmentation methods (Cheng et al., 2020; Kirillov et al., 2019a; Li et al., 2020b; Mohan and Valada, 2021; Porzi et al., 2019; Xiong et al., 2019). Part segmentation, or part parsing, on the other hand, seeks to semantically analyze the image based on part-level semantics for each class. There has been some effort in this area, but often part segmentation has been treated as a semantic segmentation problem (Gong et al., 2019; Jiang and Chi, 2018, 2019; Li et al., 2017a; Liu et al., 2018; Luo et al., 2013). There are a few instance-aware methods (Gong et al., 2018; Li et al., 2017a; Zhao et al., 2018) and even fewer that handle multi-class part objects (Zhao et al., 2019; Michieli et al., 2020).

Part-aware panoptic segmentation (de Geus et al., 2021) was recently introduced to unify semantic, instance, and part segmentation. An example of part-aware panoptic segmentation is shown in Figure 1. In (de Geus et al., 2021), a baseline approach is presented in which two networks are used, one for panoptic segmentation and the other for part segmentation. These two networks are trained independently and the results of both are combined using a uni-directional (top-down) merging strategy. This technique of independent training has significant drawbacks. Due to the use of two different networks, there is a computational overhead. As the authors employ different networks, there will be no consistency in their predictions, making the merging process inefficient. Also, the independent training strategy leads to learning redundancy since they could potentially share semantic

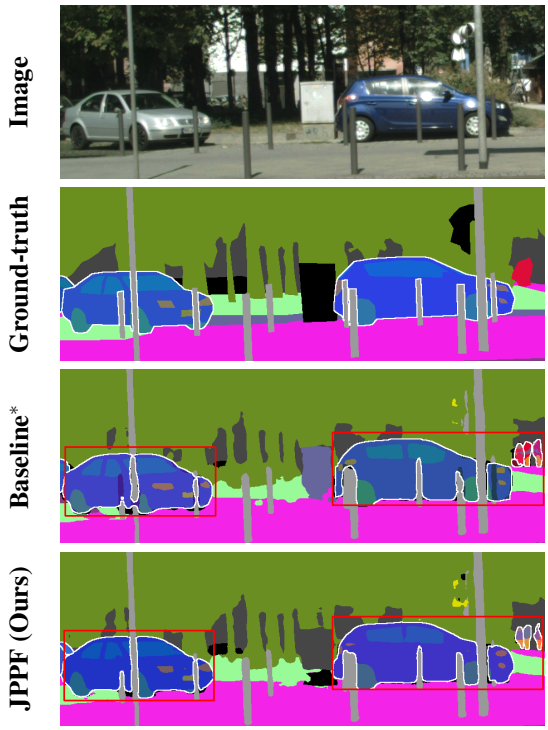


Figure 1: We propose a unified network with Joint Panoptic Part Fusion (JPPF) to generate panoptic-part segmentation. Here, a prediction of our proposed model on CPP (Meletis et al., 2020) is shown. Details about the baseline are given in Section 4.1.

information between segmentation heads.

In this work, we propose a joint network that uses a shared feature extractor to perform semantic, instance, and part segmentation. To achieve panoptic-part segmentation, we propose Joint Panoptic Part Fusion (JPPF), which fuses all three predictions by giving equal priority to each prediction head. The following is a summary of key contributions of this paper:

- We present a single new network that uses a shared encoder to perform semantic, instance, and part segmentation and fuse them efficiently to produce panoptic-part segmentation.
- To achieve panoptic-part segmentation, we propose a parameter-free joint panoptic part fusion module that dynamically considers the logits from the semantic, instance, and part head and consistently integrates the three predictions.
- We conduct a thorough analysis of our approach and demonstrate the shared encoder's efficacy and the consistency of the novel, joint fusion strategy.
- When compared to state-of-the-art (de Geus et al., 2021), our suggested fusion yields denser results at a higher quality.

2 RELATED WORK

Part-aware panoptic segmentation (de Geus et al., 2021) is a recently introduced problem that brings semantic, instance, and part segmentation together. There have been several methods proposed for these individual tasks, including panoptic segmentation, which is a blend of semantic and instance segmentation.

2.1 Towards Panoptic-Part Segmentation

Semantic Segmentation. PSPnet (Zhao et al., 2017) introduced the pyramid pooling module, which focuses on the importance of multi-scale features by learning them at many scales, then concatenating and up-sampling them. Chen et al. (2017) proposed Atrous Spatial Pyramid Pooling (ASPP), which is based on spatial pyramid pooling and combines features from several parallel atrous convolutions with varying dilation rates, as well as global average pooling. The incorporation of multi-scale characteristics and the capturing of global context increases computational complexity. So, Chen et al. (2018a) introduced the Dense Predtiction Cell (DPC) and Valada et al. (2018) suggested multi-scale residual units with changing dilation rates to compute high-resolution features at various spatial densities, as well as an efficient atrous spatial pyramid pooling module called eASPP to learn multi-scale representation with fewer parameters and a broader receptive field. In the encoder-decoder architecture, a lot of effort has been advocated for improving the decoder's upsampling layer. Chen et al. (2018b) extend DeepLabV3 (Chen et al., 2017) by adding an efficient decoder module to enhance segmentation results at object boundaries. Later, Tian et al. (2019) suggest replacing it with data-dependent up-sampling (DUpsampling), which can recover pixel-wise prediction from low-resolution CNN outputs and take advantage of the redundant label space in semantic segmentation.

Instance Segmentation. Here, we mainly concentrate on proposal based approaches. Hariharan et al. (2014) proposed a simultaneous object recognition and segmentation technique that uses Multi-scale Combinatorial Grouping (MCG) (Pont-Tuset et al., 2016) to generate proposals and then run them through a CNN for feature extraction. In addition, Hariharan et al. (2015) presented a hyper-column pixel descriptor that captures feature representations of all layers in a CNN with a strong correlation for simultaneous object detection and segmentation.

O Pinheiro et al. (2015) proposed the DeepMask network, which employs a CNN to predict the segmentation mask of each object as well as the likelihood of the object being in the patch. FCIS (Li et al., 2017b) employs position sensitive inside/outside score maps to simultaneously predict object detection and segmentation. Later, one of the most popular networks for instance segmentation, Mask-RCNN (He et al., 2017), was introduced. It extends Faster-RCNN (Ren et al., 2015) with an extra network that segments each of the detected objects. RoI-align, which preserves exact spatial position, replaces RoI-pool, which performs coarse spatial quantization for feature encoding.

Part Segmentation. Dense part level segmentation, on the other hand, is instance agnostic and is regarded as a semantic segmentation problem (Gong et al., 2019; Jiang and Chi, 2018, 2019; Li et al., 2017a; Liu et al., 2018; Luo et al., 2018; Michieli et al., 2020; Zhao et al., 2019). Most of the research has been conducted to perform human part parsing (Zhao et al., 2018; Gong et al., 2018; Dong et al., 2013; Ladicky et al., 2013; Li et al., 2020a; Liang et al., 2018; Lin et al., 2020; Ruan et al., 2019; Yang et al., 2019a), and only little work has addressed multi-part segmentation tasks (Zhao et al., 2019; Michieli et al., 2020).

Panoptic Segmentation. The authors of (Kirillov et al., 2019b) combined the output of two independent networks for semantic and instance segmentation and coined the term panoptic segmentation. Panoptic segmentation approaches can be divided into top-down methods (Li et al., 2018b; Liu et al., 2019; Li et al., 2018a; Xiong et al., 2019; Sofiiuk et al., 2019; Porzi et al., 2019) that prioritize semantic segmentation prediction and bottom-up methods (Yang et al., 2019b; Cheng et al., 2020; Gao et al., 2019) that prioritize instance prediction. In this work, we build on EfficientPS (Mohan and Valada, 2021) which will be extended to perform panoptic-part segmentation.

2.2 Panoptic-Part Segmentation.

In recent years, Part-Aware Panoptic Segmentation (de Geus et al., 2021) was introduced, which aims at a unified scene and part-parsing. Also, de Geus et al. (2021) introduced a baseline model using a state-of-the-art panoptic segmentation network and a part segmentation network, merging them using heuristics. The panoptic and part segmentation is merged in top-down or bottom-up manner. In the top-down merge, the prediction from panoptic segmentation is re-used for scene-level semantic classes that do not consist of parts. Then for partitionable semantic classes, the corresponding segment of the part prediction is extracted. In case of conflicting predictions, a void label will be assigned. According to de Geus et al. (2021), top-down merge produces better results than the bottom-up approach. In addition, their paper has released two datasets with panoptic-part annotations: Cityscapes Panoptic Part (CPP) dataset and Pascal Panoptic Part (PPP) dataset (Meletis et al., 2020). Along with the drawbacks of employing independent networks as mentioned in Section 1, there are concerns with the usage of top-down merge as shown in Figures 1 and 4. Due to inconsistencies, top-down merging may result in undefined regions around the contours of objects. Due to some imbalance between *stuff* and *things*, it also has trouble separating them. Our work resolves these issues by proposing a unified fusion for semantics, instances, and parts, giving equal priority to all individual predictions.

3 Unified Panoptic-Part Segmentation

Our work extends EfficientPS (Mohan and Valada, 2021) in two fundamental aspects: 1.) The network is extended to incorporate a part segmentation head, and 2.) we propose our joint panoptic part fusion.

3.1 Network Architecture

We employ the backbone, semantic head, and instance head of EfficientPS (Mohan and Valada, 2021) in this work. As part segmentation is regarded as a semantic segmentation problem, we are replicating the semantic branch of EfficientPS and train it for part-level segmentation. All three resulting heads share a common EfficientNet-b5 backbone (Tan and Le, 2019), which helps to ensure that the predictions made by the heads are consistent with one another. The positive impact of the shared encoder is presented in Section 4.2. In order to produce panoptic-part segmentation, we combine the predictions from all three heads in our proposed joint fusion. The goal of panoptic-part segmentation is to predict $(s, p, id)_i$ for each pixel i. Here, s represents semantic scene level class from the semantic head, p represents the part-level class and id indicates the instance identifier which is obtained from the instance head. An overview of the architecture of our proposed model is shown in Figure 2.

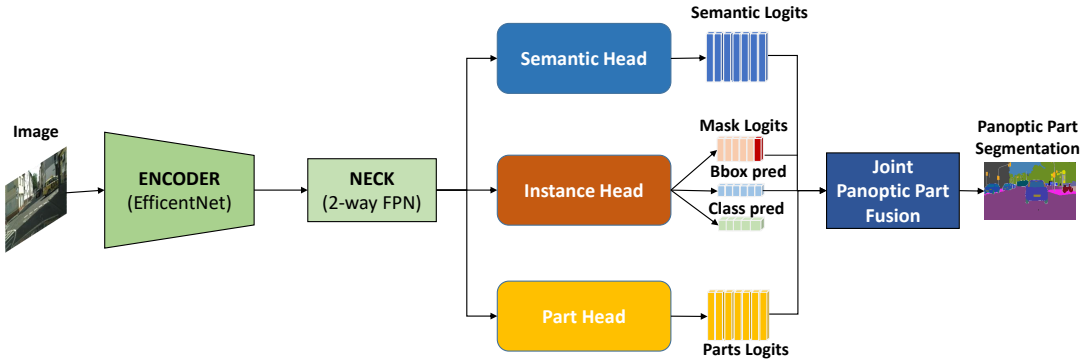


Figure 2: Overview of our proposed network architecture. It features a shared encoder, three specialized prediction heads, and a unified joint fusion module.

3.1.1 Part Segmentation Head

According to previous work (de Geus et al., 2021), the grouping of parts yields better results. I.e., semantically identical parts, e.g. the windows of cars or busses, are grouped into a single part class. We have verified this finding for our network architecture (see Table 3) and consequently follow the same principle. Another relevant design question for the part prediction head is concerned with the non-partitionable classes. In our approach, we chose to represent all these classes as a single background class. This avoids redundant predictions and further balances the learning of parts versus other classes. Our decision is again validated by experiments of which the results are provided in Table 3. Both groupings of classes (semantic grouping of parts, as well as grouping of the background) can later be easily distinguished by the additional information of the other prediction heads to obtain a fine-grained panoptic-part segmentation.

3.2 Joint Fusion

To obtain panoptic-part segmentation, one must combine the predictions of semantic segmentation, instance segmentation, and part segmentation. In general, this includes four possible categories for fusion: Partitionable and non-partitionable *stuff*, and partitionable and non-partitionable *things*. For the sake of verbosity, we only describe the three combinations which actually occur in the data (partitionable *stuff* is not included), but our approach generalized to the last case as well. Inspired by the panoptic fusion module of EfficientPS (Mohan and Valada, 2021), we propose a joint panoptic part fusion module that fuses the individual results of the three heads by giving each prediction equal priority and thoroughly exploiting coherent predictions. Figure 3 depicts our proposed joint panoptic part fusion module.

Fusion for *Things*. The instance segmentation head predicts a set of object instances, each with its class prediction, confidence score, mask logits, and bounding box prediction. The predicted instances are prefiltered according to the steps carried out by EfficientPS (Mohan and Valada, 2021), including confidence thresholding, non-maximum suppression, etc. After this, we obtain a bounding box, class prediction, and masked instance logits *MLI* for every instance. Simultaneously, we obtain the semantic logits of N channels from the semantic head, where N is the number of semantic classes, which is $N_{stuff} + N_{things}$. Lastly, we obtain the part logits with N_P channels from the part head, where N_P is the number of grouped parts plus one additional channel for the background. To balance the individual predictions, we normalize the semantic and part logits by applying a softmax function along the channel dimension. In a next step, the appropriate channels of the semantic prediction is selected, based on the class prediction of each instance. This selected logits are further masked according to the instance's bounding box to yield the masked, semantic logits *MLS*.

Suppose the predicted class (by the instance head) is partitionable, then a subset of corresponding logits are selected from the part segmentation, e.g. if the instance head predicts a person, the logits for head, torso, legs, and arms are selected. These logits are again masked by the corresponding bounding box to produce the third masked logits for parts *MLP*. If the predicted class is not further segmentable into parts, the background class from the part logits is selected instead and masked likewise. To make the fusion operation feasible, we replicate *MLS* and *MLI* to match the number of corresponding parts. For example, a person instance contains four parts (head, arms, torso, legs), thus *MLP* is of shape $4 \times W \times H$. Therefore, *MLS* and *MLI* are replicated 4 times to match the shape of *MLP*.

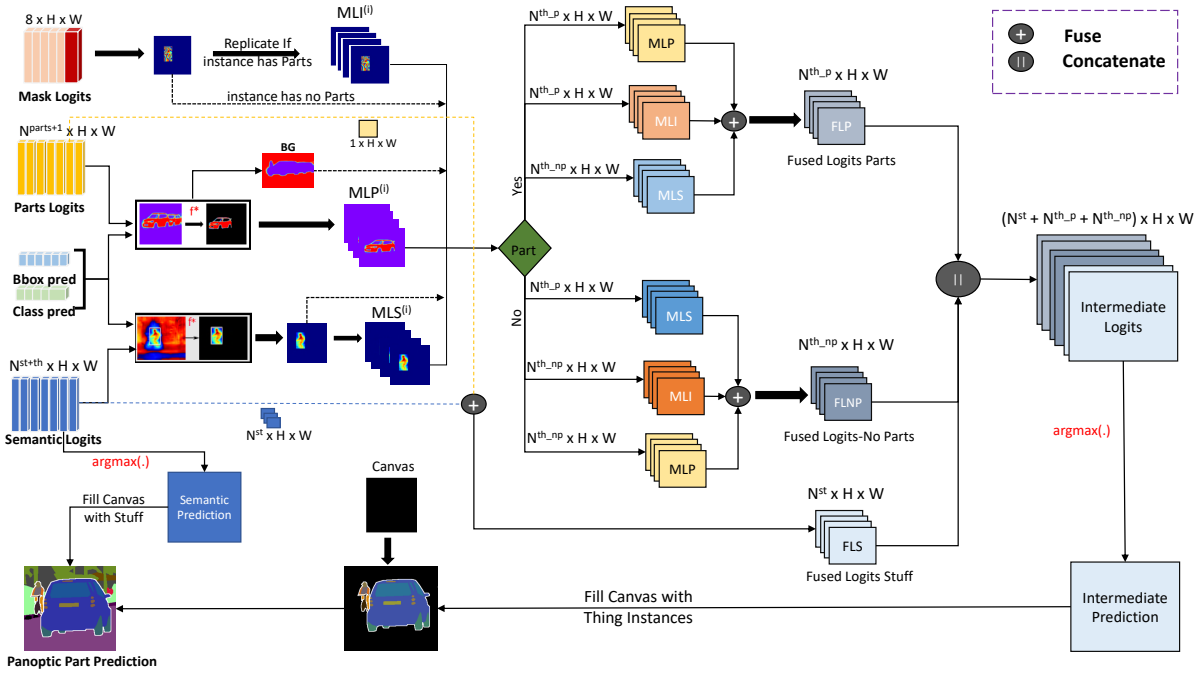


Figure 3: Illustration of our proposed joint fusion module. Semantic, instance, and part predictions are equally balanced and combined.

By now, three sets of masked logits are available. We are now fusing these logits separately for classes with and without parts in the same fashion. To compute the fused logits for classes with parts FLP and class without parts $FLNP$, we form the sum of the sigmoid of the masked logits and the sum of the masked logits and compute the Hadamard product of both. This procedure is depicted in Equation 1:

$$FL(MLL) = \left(\sum_{l \in MLL} \sigma(l) \right) \odot \left(\sum_{l \in MLL} l \right) \quad (1)$$

In this equation, $\sigma(\cdot)$ denotes the sigmoid function, $\odot$ denotes the Hadamard product, and MLL is a set of masked logits which are supposed to be fused, e.g. $MLL = \{MLS, MLI, MLP\}$. This equation describes a generalized version of the fusion proposed by Mohan and Valada (2021) that handles arbitrarily many logits.

Fusion for *Stuff*. To generate the fused logits FLS for the *stuff* classes, the N_{stuff} channels from the semantic head are fused with the background channel of the part head in the same manner, i.e. according to Equation 1, but this time with only two sets of logits (no instance information). As mentioned, the same concept would also apply for *stuff* that is partitionable.

Overall Fusion. All three fused logits, FLP, $FLNP$, and FLS, are concatenated along the channel dimensions to obtain intermediate logits, which produce the intermediate panoptic-part prediction by taking the *argmax* of these intermediate logits. Finally, we fill an empty canvas with the intermediate panoptic-part prediction for all *things*. The remaining empty parts, i.e. the background, of the canvas is filled with the prediction for *stuff* classes extracted from the semantic segmentation head. Lastly, *stuff* areas below a minimum threshold min_{stuff} are filtered, as by Mohan and Valada (2021). During fusion, the fused score increases if the predictions of all three heads are consistent, and likewise it is decreased if the predictions do not match with eachother.

4 Experiments and Results

Datsets. As mentioned before, we use the recently introduced Cityscapes Panoptic Parts (CPP) and Pascal Panoptic Parts (PPP) datasets (Meletis et al., 2020). CPP provides pixel-level annotations for 11 *stuff* classes and 8 *things* classes, totaling 19 object classes. Out of the 8 *things*, five include annotations at the part level. There are 2975 images for training and 500 for validation in this finely annotated dataset. PPP consists of 100 object classes, with 20 *things* and 80 *stuff* classes. Part-level annotations are present in

16 of the 20 *things*. As in previous work (Meletis et al., 2020), we only consider a subset of 59 object classes for training and evaluation, including 20 *things* and 39 *stuff* classes, and 58 part classes. These parts are detailed by Michieli et al. (2020) and Zhao et al. (2019). PPP consists of a total of 10103 images which are divided into 4998 images for training and 5105 for validation.

Training Details. For the Cityscapes data, we use images of the original resolution, i.e. 1024×2048 pixels, and resize the input images of PPP to 384×512 pixels for training. We perform data augmentation, scaling and hyperparameter initialization as in EfficientPS (Mohan and Valada, 2021). We use a multi-step learning rate (lr) and train our network by Stochastic Gradient Descent (SGD) with a momentum of 0.9. For the CPP and PPP, we use a start lr of 0.07 and 0.01, respectively. We begin the training with a warm-up phase in which the lr is increased linearly from $\frac{1}{3}lr$ up to lr within 200 iterations. The weights of all InPlace-ABN layers (Bulo et al., 2018) are frozen, and we train the model for 10 additional epochs with a fixed learning rate of 10^{-4}. Finally, we unfreeze the weights of the InPlace-ABN layers and train the model for $50k$ iterations beginning with lr of 0.07 (CPP) and 0.01 (PPP), and reduce lr after iterations $32k$ and $44k$ by a factor of 10. Four GPUs are used for the training with a batch size of 2 per GPU for CPP and 8 per GPU for PPP.

Metrics. In this paper, we evaluate the individual semantic and part segmentation using mean Intersection over Union (mIoU), and the instance segmentation using mean Average Precision (mAP). For the evaluation of our panoptic-part segmentation, we use the Part Panoptic Quality (PartPQ) (de Geus et al., 2021), which is an extension of the Panoptic Quality (PQ) (Kirillov et al., 2019a).

4.1 Comparison to State-of-the-Art

The baseline approach by de Geus et al. (2021) uses the panoptic labels of the Cityscapes dataset (Cordts et al., 2016) to train a panoptic segmentation network. Since this data is slightly different from the recently annotated panoptic part dataset (CPP) presented by de Geus et al. (2021), a direct, fair comparison is not possible. Table 1 clearly demonstrates that the CPP dataset differs, as the introduction of parts has resulted in inconsistencies of annotations. To make the baseline comparable to our approach in terms of data, we re-implement the baseline and train it on the same data. The re-implementation consists of EfficientPS (Mohan and Valada, 2021) for panoptic segmentation, and our part segmentation network with a separate backbone (cf. Section 3.1.1). Top-down merging is then used to combine the two independent results into a panoptic-part segmentation. Our model is compared to the reproduced baseline and the official baseline of de Geus et al. (2021). The official baseline consists of EfficientPS (Mohan and Valada, 2021) and BSANet (Zhao et al., 2019) with top-down merging. The results of this comparison are shown in Table 2 for single-scale and multi-scale inference.

Table 1: Comparison of EfficientPS (Mohan and Valada, 2021) trained on cityscapes panoptic dataset with EfficientPS trained with Cityscapes Panoptic Part (CPP) dataset (de Geus et al., 2021) and single-scale testing. * indicates the model trained with CPP dataset.

Method	PQ	SQ	RQ
EfficientPS	63.9	81.5	77.1
EfficientPS*	62.2	81.0	75.7

For CPP, the results indicate that our proposed network improves accuracy significantly compared to the reproduced baseline for single-scale testing. Our JPPF outperforms the reproduced baseline by 1.9 percentage points (pp) in overall $PartPQ$ and significantly by 3.5 pp in $PartPQ_P$. Similarly for multi-scale testing, our proposed model outperforms the baseline by 1.6 pp and 4.7 pp in $PartPQ$ and $PartPQ_P$, respectively. Furthermore, our model betters both baselines in all individual predictions before merging/fusion. In addition, JPPF produces denser results than the baseline, which enhances the density by 0.5 pp for single-scale testing and by 0.66 pp for multi-scale testing.

For PPP, our model outperforms the top-down combination DeepLabV3+ (Chen et al., 2018b) and Mask RCNN (He et al., 2017) (*Baseline-1*), even though this baseline was trained with the original Pascal parts and Pascal panoptic segmentation datasets, which provide more annotations. *Baseline-2* (top-down merging of DLv3-ResNeSt269 (Chen et al., 2017; Zhang et al., 2022), DetectoRS (Qiao et al., 2020), and BSANet (Zhao et al., 2019)) obtains an even better result because it is constructed from much more complex models, and hence has a higher representational capacity. However, when comparing the model size (see Table 2), it shows that the backbone of *Baseline-2* alone is already more than two times larger than our whole model.

From Figure 4, we can see that our proposed fusion is able to segment the parts of very small and distant object classes reliably. Also, our proposed fusion solves the typical problems of top-down merging (cf. Section 1). As illustrated in Figure 4, there are no unknown regions within objects (*things*), since our

Table 2: Evaluation results of panoptic-part segmentation on Cityscapes and Pascal Panoptic Parts (Meletis et al., 2020) compared to state-of-the art. *P* and *NP* refer to areas with and without part labels, respectively. * indicates our reproduced baseline (details in Section 4.1). † indicates that the number of parameters refer to the encoders only.

Method	Before Merge/Fusion			After Merge/Fusion			Density	Run	Model
	Sem.	Inst.	Part	PartPQ			[%]	time	size
	mIoU	AP	mIoU	All	P	NP		[ms]	[M]
Cityscapes Panoptic Parts, Single-Scale									
Baseline*	79.7	36.6	74.5	57.7	44.2	62.5	98.84	871	68.8
JPPF (Ours)	80.5	37.9	77.0	59.6	47.7	63.8	99.33	397	44.19
Cityscapes Panoptic Parts, Multi-Scale									
Baseline	80.3	39.7	76.0	60.2	46.1	65.2	–	–	–
JPPF (Ours)	81.8	41.3	78.5	61.8	50.8	65.7	99.50	2498	44.19
Pascal Panoptic Parts, Single-Scale									
Baseline-1	47.1	38.5	53.9	31.4	47.2	26.0	–	–	68†
Baseline-2	55.1	44.8	58.6	38.3	51.6	33.8	–	–	111†
JPPF (Ours)	46.0	39.1	54.4	32.3	48.3	26.9	92.10	146	44.19

Table 3: Ablation Study on Cityscapes Panoptic Parts. The design choices of our part segmentation head are validated, and we contrast independent and shared feature encoders.

Method	Sem. mIoU	Inst. AP	Part mIoU
Grouped Parts	–	–	74.5
Non-Grouped Parts	–	–	65.7
Grouped Parts + SemBG	–	–	75.6
Grouped Parts + BG (Ours)	–	–	77.0
Independent Networks	78.1	37.3	74.5
Shared Features (Ours)	80.5	37.9	77.0

fusion gives equal priority to all three heads. The second issue of *stuff* classes bifurcating *things* (as shown in Figure 1) is also improved largely. This is due to the introduction of fusion between *stuff* classes of semantic logits and the background class of part logits. Lastly when comparing the model sizes and inference times, we can highlight another advantage of our unified model: It is more efficient as it requires fewer parameters. On average, the inference per image requires only 397ms, which is less than half of the time required by the baseline.

4.2 Ablation Study

4.2.1 Shared Encoder vs. Independent Encoders

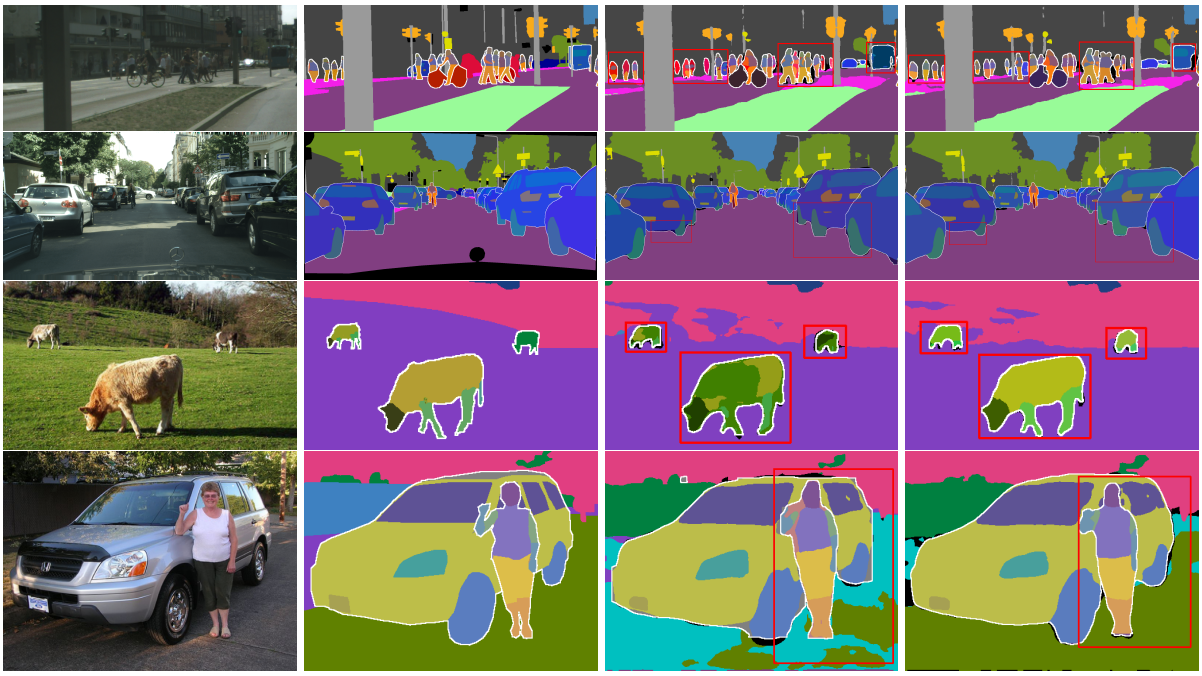
Our aim is to jointly learn semantic, instance, and part segmentation in a single, unified model. To validate that these three tasks benefit from a common feature representation, we compare our results before fusion to three separate equivalent networks that have been trained individually with different encoders. The model with a single, shared encoder surpasses the individual models in all three tasks (see Table 3). The improvement is 2.4 pp, 2.5 pp, and 0.6 pp for semantic, part, and instance segmentation, respectively. This result clearly indicates that using a shared encoder enables the network to learn a common feature representation, resulting in more accurate individual outcomes of each head.

4.2.2 Top-down Merge vs. Joint Fusion

Next, we compare our joint fusion module to the previously presented top-down merging strategy (de Geus et al., 2021) in Table 4. The proposed fusion module surpasses the top-down merge in terms of $PartPQ$, $PartPQ_P$, $PartPQ_{NP}$ in all test settings. Even though our proposed fusion is admittedly only slightly better, the joint fusion produces also denser results than the uni-directional merge, indicating the improved consistency before and after fusion. Additionally and as explained earlier, our fusion resolves the typical issues that are present with the top-down merge, as seen in Figures 1 and 4. This is achieved by incorporating the part prediction into a mutual fusion, and mainly reflected for the results in areas that are partitionable. Since the *things* with part labels are limited in CPP, the impact is best observed on the PPP dataset. On this data, our proposed fusion module is significantly better. Specifically $PartPQ_P$ is improved by 10.5 pp, by giving equal priority to the parts during fusion.

4.3 Run-time Analysis

We further assessed the efficiency of our proposed model with joint fusion, and the results are displayed in Table 5. It is evident that the top-down merging re-



Original Image | Ground-truth | Baseline* | JPPF (Ours)

Figure 4: Qualitative results of our proposed model on Citscapes and Pascal Panoptic Parts compared to our reproduced baseline, ground-truth and the reference image. More visual examples for both datasets are provided in the appendix in Figures 5 and 6.

Table 4: Ablation Study on Cityscapes and Pascal Panoptic Parts (Meletis et al., 2020). We compare the uni-directional top-down merge to our proposed joint fusion module.

Method	Before Merge/Fusion Sem. mIoU	Inst. AP	Part mIoU	After Merge/Fusion PartPQ All	P	NP	Density [%]
Cityscapes Panoptic Parts, Single-Scale							
Ours w/ Top-Down-Merge	80.5	37.9	77.0	59.5	47.5	63.7	99.13
JPPF (Ours)	80.5	37.9	77.0	59.6	47.7	63.8	99.33
Cityscapes Panoptic Parts, Multi-Scale							
Ours w/ Top-Down-Merge	81.8	41.3	78.5	61.6	50.7	65.5	99.20
JPPF (Ours)	81.8	41.3	78.5	61.8	50.8	65.7	99.50
Pascal Panoptic Parts, Single-Scale							
Ours w/ Top-Down-Merge	46.0	39.1	54.4	29.0	37.8	26.0	89.57
JPPF (Ours)	46.0	39.1	54.4	32.3	48.3	26.9	92.10

Table 5: Run-time comparison of JPPF to the baseline on Cityscapes Panoptic Parts. * indicates the reproduced baseline which is detailed in Section 4.1.

Method	Individual Predictions [ms]	Fuse/Merge [ms] Panoptic Fusion	Merge	Joint Fusion	Total Inference [ms]
Baseline*	269	118	484	–	871
Ours w/ Merge	215	118	484	–	817
JPPF (Ours)	236	–	–	161	397

quires more than twice the time compared to our proposed fusion. To obtain panoptic-part segmentation as proposed by de Geus et al. (2021), one must first perform a panoptic fusion and then combine it with the part segmentation, which adds an extra overhead. In comparison to the baseline, our approach is even more efficient because it uses a single backbone.

5 Conclusion

In this paper, we proposed a unified network that helps to generate semantic, instance, and part segmentation and effectively combines them to provide a consistent panoptic-part segmentation. Our proposed model with joint fusion significantly outperforms the state-of-the-art by 1.6 pp in overall *PartPQ* and by 4.7 pp in $PartPQ_P$ on the CPP dataset. For the PPP dataset, our model with joint fusion outperforms our model with the top-down merge significantly by 3.3 pp in overall *PartPQ* and by 10.5 pp in $PartPQ_P$. With the addition of *stuff* and parts into the fusion, our suggested fusion modules addresses the problems encountered in the top-down merge, such as unknown pixels inside contours and the bifurcation of *things* and *stuff*. When compared to top-down merge, our suggested joint fusion is faster and produces denser results with superior segmentation quality.

For future work, we plan to interpolate the remaining, filtered regions in the prediction to obtain a fully dense panoptic-part segmentation.

ACKNOWLEDGEMENTS

This work was partially funded by the Federal Ministry of Education and Research Germany under the project DECODE (01IW21001).

REFERENCES

Bulo, S. R., Porzi, L., and Kontschieder, P. (2018). In-place activated batchnorm for memory-optimized training of dnns. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 6

Chen, L., Collins, M. D., Zhu, Y., Papandreou, G., Zoph, B., Schroff, F., Adam, H., and Shlens, J. (2018a). Searching for efficient multi-scale architectures for dense image prediction. *Advances in Neural Information Processing Systems (NeurIPS)*. 2

Chen, L., Papandreou, G., Schroff, F., and Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*. 2, 6

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018b). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European conference on computer vision (ECCV)*. 2, 6

Cheng, B., Collins, M. D., Zhu, Y., Liu, T., Huang, T. S., Adam, H., and Chen, L.-C. (2020). Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 1, 3

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 1, 6

de Geus, D., Meletis, P., Lu, C., Wen, X., and Dubbelman, G. (2021). Part-aware panoptic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 1, 2, 3, 4, 6, 7, 9

Dong, J., Chen, Q., Xia, W., Huang, Z., and Yan, S. (2013). A deformable mixture parsing model with parselets. In *International Conference on Computer Vision (ICCV)*. 3

Gao, N., Shan, Y., Wang, Y., Zhao, X., Yu, Y., Yang, M., and Huang, K. (2019). Ssap: Single-shot instance segmentation with affinity pyramid. In *International Conference on Computer Vision (ICCV)*. 3

Gong, K., Gao, Y., Liang, X., Shen, X., Wang, M., and Lin, L. (2019). Graphonomy: Universal human parsing via graph transfer learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 1, 3

Gong, K., Liang, X., Li, Y., Chen, Y., Yang, M., and Lin, L. (2018). Instance-level human parsing via part grouping network. In *European Conference on Computer Vision (ECCV)*. 1, 3

Hariharan, B., Arbeláez, P., Girshick, R., and Malik, J. (2014). Simultaneous detection and segmentation. In *European Conference on Computer Vision (ECCV)*. 2

Hariharan, B., Arbeláez, P., Girshick, R., and Malik, J. (2015). Hypercolumns for object segmentation and fine-grained localization. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask r-cnn. In *International Conference on Computer Vision (ICCV)*. 3, 6

Jiang, Y. and Chi, Z. (2018). A cnn model for semantic person part segmentation with capacity optimization. *Transactions on Image Processing (T-IP)*. 1, 3

Jiang, Y. and Chi, Z. (2019). A cnn model for human parsing based on capacity optimization. *Applied Sciences*. 1, 3

Kirillov, A., Girshick, R., He, K., and Dollár, P. (2019a). Panoptic feature pyramid networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 1, 6

Kirillov, A., He, K., Girshick, R., Rother, C., and Dollár, P. (2019b). Panoptic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 1, 3

Ladicky, L., Torr, P. H., and Zisserman, A. (2013). Human pose estimation using a joint pixel-wise and part-wise formulation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 3

Li, J., Raventos, A., Bhargava, A., Tagawa, T., and Gaidon, A. (2018a). Learning to fuse things and stuff. *arXiv preprint arXiv:1812.01192*. 3

Li, P., Xu, Y., Wei, Y., and Yang, Y. (2020a). Self-correction for human parsing. *Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*. 3

Li, Q., Arnab, A., and Torr, P. H. (2017a). Holistic, instance-level human parsing. *British Machine Vision Conference (BMVC)*. 1, 3

Li, Q., Arnab, A., and Torr, P. H. (2018b). Weakly-and semi-supervised panoptic segmentation. In *European conference on computer vision (ECCV)*. 3

Li, Q., Qi, X., and Torr, P. H. (2020b). Unifying training and inference for panoptic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 1

Li, Y., Qi, H., Dai, J., Ji, X., and Wei, Y. (2017b). Fully convolutional instance-aware semantic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 3

Liang, X., Gong, K., Shen, X., and Lin, L. (2018). Look into person: Joint body parsing & pose estimation network and a new benchmark. *Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*. 3

Lin, K., Wang, L., Luo, K., Chen, Y., Liu, Z., and Sun, M.-T. (2020). Cross-domain complementary learning using pose for multi-person part segmentation. *Transactions on Circuits and Systems for Video Technology (T-CSVT)*. 3

Liu, H., Peng, C., Yu, C., Wang, J., Liu, X., Yu, G., and Jiang, W. (2019). An end-to-end network for panoptic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 3

Liu, S., Sun, Y., Zhu, D., Ren, G., Chen, Y., Feng, J., and Han, J. (2018). Cross-domain human parsing via adversarial feature and label adaptation. In *Conference On Artificial Intelligence (AAAI)*. 1, 3

Luo, P., Wang, X., and Tang, X. (2013). Pedestrian parsing via deep decompositional network. In *International Conference on Computer Vision (ICCV)*. 1

Luo, X., Su, Z., Guo, J., Zhang, G., and He, X. (2018). Trusted guidance pyramid network for human parsing. In *ACM International Conference on Multimedia (ACM-MM)*. 3

Meletis, P., Wen, X., Lu, C., de Geus, D., and Dubbelman, G. (2020). Cityscapes-panoptic-parts and pascal-panoptic-parts datasets for scene understanding. *arXiv preprint arXiv:2004.07944*. 2, 3, 5, 6, 7, 8, 11, 12

Michieli, U., Borsato, E., Rossi, L., and Zanuttigh, P. (2020). Gmnet: Graph matching network for large scale part semantic segmentation in the wild. In *European Conference on Computer Vision (ECCV)*. 1, 3, 6

Mohan, R. and Valada, A. (2021). EfficientPS: Efficient Panoptic Segmentation. *International Journal of Computer Vision (IJCV)*. 1, 3, 4, 5, 6

O Pinheiro, P. O., Collobert, R., and Dollár, P. (2015). Learning to segment object candidates. *Advances in Neural Information Processing Systems (NeurIPS)*. 3

Pont-Tuset, J., Arbelaez, P., Barron, J. T., Marques, F., and Malik, J. (2016). Multiscale combinatorial grouping for image segmentation and object proposal generation. *Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*. 2

Porzi, L., Bulo, S. R., Colovic, A., and Kontschieder, P. (2019). Seamless scene segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 1, 3

Qiao, S., Chen, L.-C., and Yuille, A. (2020). Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. *arXiv preprint arXiv:2006.02334*. 6

Ren, S., He, K., Girshick, R. B., and Sun, J. (2015). Faster R-CNN: towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems (NeurIPS)*. 3

Ruan, T., Liu, T., Huang, Z., Wei, Y., Wei, S., and Zhao, Y. (2019). Devil in the details: Towards accurate single and multiple human parsing. In *Conference on Artificial Intelligence (AAAI)*. 3

Sofiiuk, K., Barinova, O., and Konushin, A. (2019). Adaptis: Adaptive instance selection network. In *International Conference on Computer Vision (ICCV)*. 3

Tan, M. and Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning (ICML)*. 3

Tian, Z., He, T., Shen, C., and Yan, Y. (2019). Decoders matter for semantic segmentation: Data-dependent decoding enables flexible feature aggregation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2

Valada, A., Mohan, R., and Burgard, W. (2018). Self-supervised model adaptation for multimodal semantic segmentation. *International Journal of Computer Vision (IJCV)*. 2

Xiong, Y., Liao, R., Zhao, H., Hu, R., Bai, M., Yumer, E., and Urtasun, R. (2019). Upsnet: A unified panoptic segmentation network. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 1, 3

Yang, L., Song, Q., Wang, Z., and Jiang, M. (2019a). Parsing r-cnn for instance-level human analysis. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 3

Yang, T., Collins, M. D., Zhu, Y., Hwang, J., Liu, T., Zhang, X., Sze, V., Papandreou, G., and Chen, L. (2019b). Deeperlab: Single-shot image parser. *arXiv preprint arXiv:1902.05093*. 3

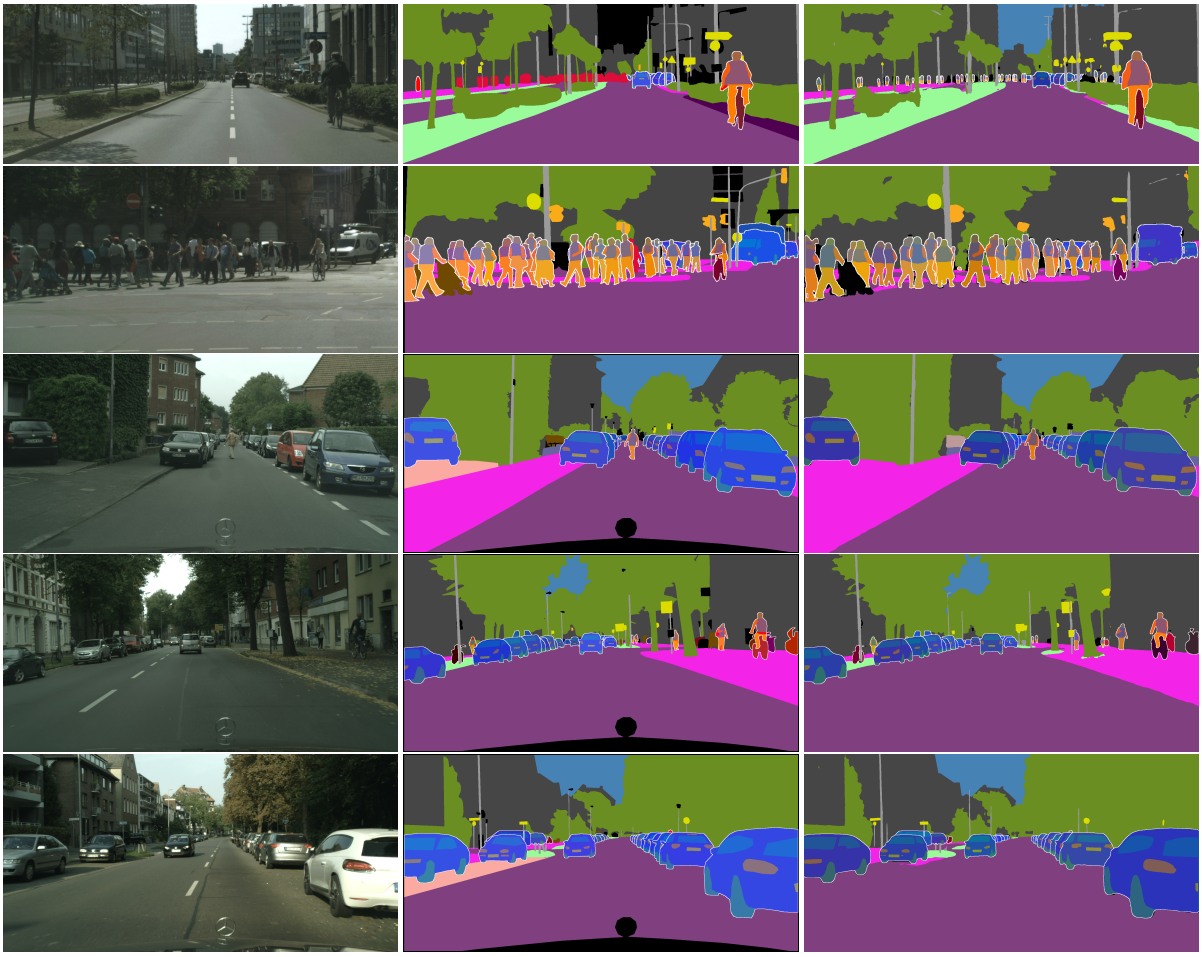
Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., Sun, Y., He, T., Mueller, J., Manmatha, R., et al. (2022). Resnest: Split-attention networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 6

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid scene parsing network. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2

Zhao, J., Li, J., Cheng, Y., Sim, T., Yan, S., and Feng, J. (2018). Understanding humans in crowded scenes: Deep nested adversarial learning and a new benchmark for multi-human parsing. In *ACM International Conference on Multimedia (ACM-MM)*. 1, 3

Zhao, Y., Li, J., Zhang, Y., and Tian, Y. (2019). Multi-class part parsing with joint boundary-semantic awareness. In *International Conference on Computer Vision (ICCV)*. 1, 3, 6

APPENDIX



Original Image Ground-truth JPPF (Ours)

Figure 5: Qualitative results of our proposed model compared to the ground truth and the reference image on CPP (Meletis et al., 2020).

Original Image Ground-truth JPPF (Ours)

Figure 6: Qualitative results of our proposed model compared to the ground truth and the reference image on PPP (Meletis et al., 2020).