

OneFormer: One Transformer to Rule Universal Image Segmentation

Jitesh Jain^{1,2}, Jiachen Li^{*}, MangTik Chiu^{*}, Ali Hassani¹, Nikita Orlov³, Humphrey Shi^{1,3}

¹SHI Labs @ U of Oregon & UIUC, ²IIT Roorkee, ³Picsart AI Research (PAIR)

<https://github.com/SHI-Labs/OneFormer>

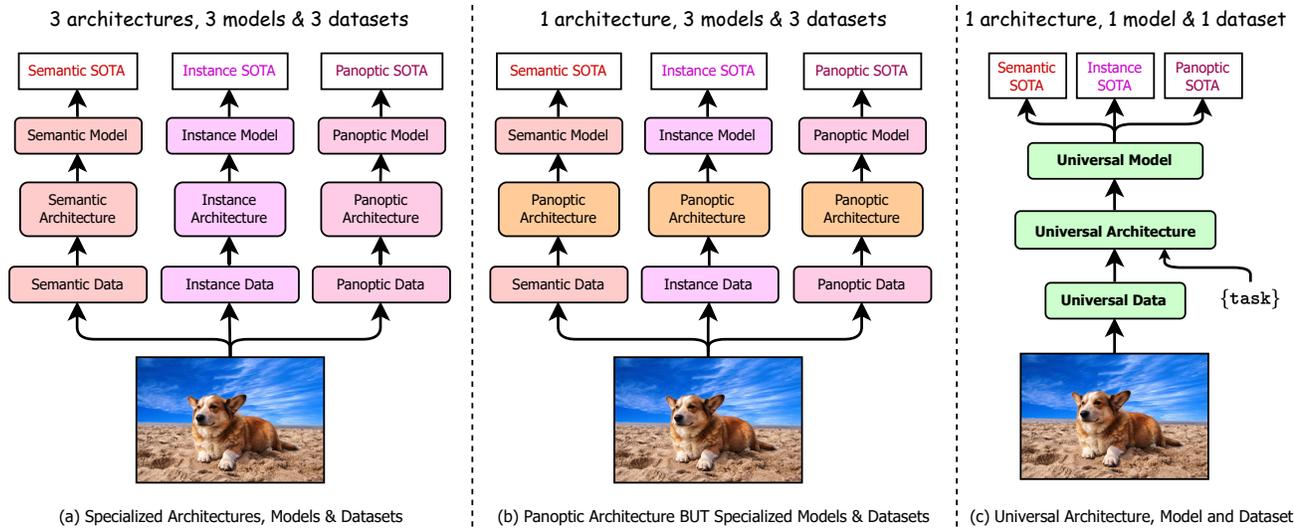


Figure 1. **A Path to Universal Image Segmentation.** (a) Traditional segmentation methods developed specialized architectures and models for each task to achieve top performance. (b) Recently, new panoptic/universal architectures [10, 47] used the same architecture to achieve top performance across different tasks. However, they still need to train different models for different tasks, resulting in a semi-universal approach. (c) We propose a unique multi-task universal architecture with a task-conditioned joint training strategy that sets new state-of-the-arts across semantic, instance and panoptic segmentation tasks with a single model, unifying segmentation across architecture, model and dataset. Our work significantly reduces the underlying resource requirements, making segmentation more universal and accessible.

Abstract

Universal Image Segmentation is not a new concept. Past attempts to unify image segmentation include scene parsing, panoptic segmentation, and, more recently, new panoptic architectures. However, such panoptic architectures do not truly unify image segmentation because they need to be trained individually on the semantic, instance, or panoptic segmentation to achieve the best performance. Ideally, a truly universal framework should be trained only once and achieve SOTA performance across all three image segmentation tasks. To that end, we propose OneFormer, a universal image segmentation framework that unifies segmentation with a multi-task train-once design. We first propose a task-conditioned joint training strategy that enables training on ground truths of each domain (semantic, instance, and panoptic segmentation) within a single multi-task training process. Secondly, we introduce a task token to condition our model on the task at hand, making our model task-dynamic to support multi-task training and inference.

Thirdly, we propose using a query-text contrastive loss during training to establish better inter-task and inter-class distinctions. Notably, our single OneFormer model outperforms specialized Mask2Former models across all three segmentation tasks on ADE20k, Cityscapes, and COCO, despite the latter being trained on each task individually. We believe OneFormer is a significant step towards making image segmentation more universal and accessible.

1. Introduction

Image Segmentation is the task of grouping pixels into multiple segments. Such grouping can be semantic-based (e.g., road, sky, building), or instance-based (objects with well-defined boundaries). Earlier segmentation approaches [6, 19, 32] tackled these two segmentation tasks individually, with specialized architectures and therefore separate research effort into each. In a recent effort to unify semantic and instance segmentation, Kirillov *et al.* [23] proposed panoptic segmentation, with pixels grouped into an

amorphous segment for amorphous background regions (labeled “stuff”) and distinct segments for objects with well-defined shape (labeled “thing”). This effort, however, led to new specialized panoptic architectures [9] instead of unifying the previous tasks (see Fig. 1a). More recently, the research trend shifted towards unifying image segmentation with new panoptic architectures, such as K-Net [47], MaskFormer [11], and Mask2Former [10]. Such panoptic/universal architectures can be trained on all three tasks and obtain high performance without changing architecture. They do need to, however, be trained individually on each task to achieve the best performance (see Fig. 1b). The individual training policy requires extra training time and produces different sets of model weights for each task. In that regard, they can only be considered a semi-universal approach. For example, Mask2Former [10] is trained for 160K iterations on ADE20K [13] for each of the semantic, instance, and panoptic segmentation tasks to obtain the best performance for each task, yielding a total of 480k iterations in training, and three models to store and host for inference.

In an effort to truly unify image segmentation, we propose a multi-task universal image segmentation framework (**OneFormer**), which outperforms existing state-of-the-arts on all three image segmentation tasks (see Fig. 1c), by only training once on one panoptic dataset. Through this work, we aim to answer the following questions:

(i) *Why are existing panoptic architectures [10, 11] not successful with a single training process or model to tackle all three tasks?* We hypothesize that existing methods need to train individually on each segmentation task due to the absence of task guidance in their architectures, making it challenging to learn the inter-task domain differences when trained jointly or with a single model. To tackle this challenge, we introduce a task input token in the form of text: “the task is {task}”, to condition the model on the task in focus, making our architecture task-guided for training, and task-dynamic for inference, all with a single model. We uniformly sample {task} from {panoptic, instance, semantic} and the corresponding ground truth during our joint training process to ensure our model is unbiased in terms of tasks. Motivated by the ability of panoptic [23] data to capture both semantic and instance information, we derive the semantic and instance labels from the corresponding panoptic annotations during training. Consequently, we only need panoptic data during training. Moreover, our joint training time, model parameters, and FLOPs are comparable to the existing methods, decreasing training time and storage requirements up to 3×, making image segmentation less resource intensive and more accessible.

(ii) *How can the multi-task model better learn inter-task and inter-class differences during the single joint training process?* Following the recent success of transformer frameworks [2, 10, 17, 18, 21, 30, 46] in computer vision, we formu-

late our framework as a transformer-based approach, which can be guided through the use of query tokens. To add task-specific context to our model, we initialize our queries as repetitions of the task token (obtained from the task input) and compute a query-text contrastive loss [33, 43] with the text derived from the corresponding ground-truth label for the sampled task as shown in Fig. 2. We hypothesize that a contrastive loss on the queries helps guide the model to be task-sensitive and reduce category mispredictions.

We evaluate OneFormer on three major segmentation datasets: ADE20K [13], Cityscapes [12], and COCO [27], each with all three segmentation tasks. OneFormer sets the new state of the arts for all three tasks with a single jointly trained model. To summarize, our main contributions are:

- We propose OneFormer, the first transformer-based multi-task universal image segmentation framework that needs to be trained only once with a single universal architecture, a single model, and on a single dataset to outperform existing frameworks across the semantic, instance, and panoptic segmentation tasks, despite the latter need to be trained separately on each task.
- OneFormer uses a task-conditioned joint training strategy, uniformly sampling different ground truth domains (semantic, instance, or panoptic) by deriving all GT labels from panoptic annotations to train its multi-task model. Thus, OneFormer actually achieves the original unification goal of panoptic segmentation [23].
- We validate OneFormer through extensive experiments on three major benchmarks: ADE20K [13], Cityscapes [12], and COCO [27]. OneFormer sets a new state-of-the-art performance on all three segmentation tasks compared with methods using the standard Swin-L [30] backbone and improves even more with new ConvNeXt [31] and DiNAT [17] backbones.

2. Related Work

2.1. Image Segmentation

Image segmentation is one of the most fundamental tasks in image processing and computer vision. Traditional works usually tackle one of the three image segmentation tasks with specialized network architectures (Fig. 1a).

Semantic Segmentation. Semantic segmentation was long tackled as a pixel classification problem with CNNs [5, 6, 8, 20, 32]. More recent works [21, 34, 42] have shown the success of transformer-based methods in semantic segmentation following its success in language and vision [2, 37]. Among them, MaskFormer [11] treated semantic segmentation as a mask classification problem following early works [3, 14, 16], through using a transformer decoder with object queries [2]. We also formulate semantic segmentation as a mask classification problem.

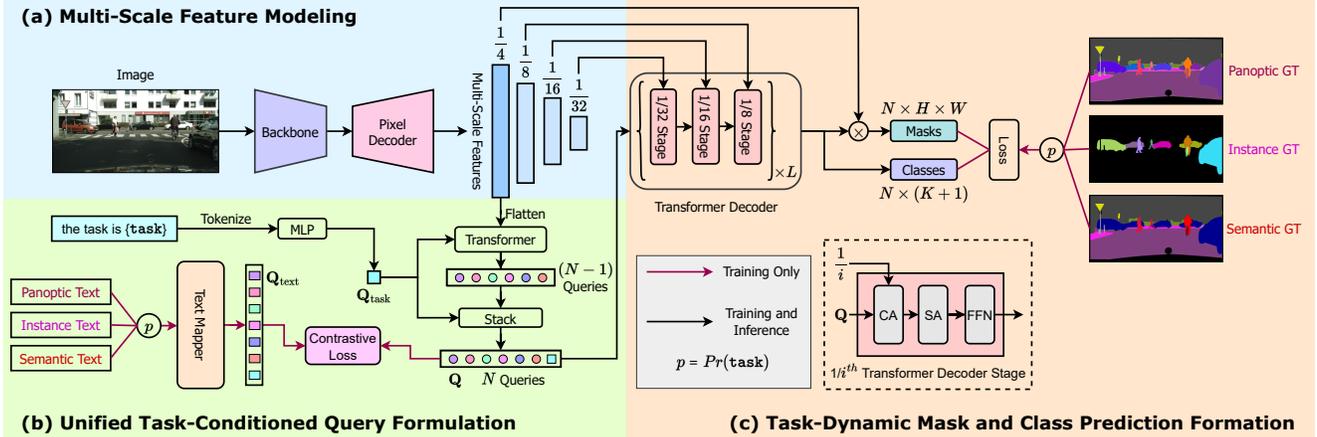


Figure 2. **OneFormer Framework Architecture.** (a) We extract multi-scale features for an input image using a backbone, followed by a pixel decoder. (b) We formulate a unified set of $N - 1$ task-conditioned object queries with guidance from the task token (\mathbf{Q}_{task}) and flattened 1/4-scale features inside a transformer [37]. Next, we concatenate \mathbf{Q}_{task} with the $N - 1$ queries from the transformer. We uniformly ($p = 1/3$) sample the task during training and generate the corresponding text queries (\mathbf{Q}_{text}) using a text mapper (Fig. 4). We calculate a query-text contrastive loss to learn the inter-task distinctions. We can drop the text mapper during inference, thus, making our model parameter efficient. (c) We use a multi-stage L -layer transformer decoder to obtain the task-dynamic class and mask predictions.

Instance Segmentation. Traditional instance segmentation methods [1, 4, 19] are also formulated as mask classifiers, which predict binary masks and a class label for each mask. We also formulate instance segmentation as a mask classification problem.

Panoptic Segmentation. Panoptic Segmentation [23] was proposed to unify instance and semantic segmentation. One of the earliest architectures in this scope was Panoptic-FPN [22], which introduced separate instance and semantic task branches. Works that followed significantly improved performance with transformer-based architectures [10, 11, 38, 39, 45, 46]. Despite the progress made so far, panoptic segmentation models are still behind in performance compared to individual instance and semantic segmentation models, therefore not living up to their full unification potential. Motivated by this, we design our OneFormer to be trained with panoptic annotations only.

2.2. Universal Image Segmentation

The concept of universal image segmentation has existed for some time, starting with image and scene parsing [35, 36, 44], followed by panoptic segmentation [23]. More recently, promising architectures [10, 11, 47] designed specifically for panoptic segmentation have emerged which also perform well on semantic and instance segmentation tasks. K-Net [47], a CNN, uses dynamic learnable instance and semantic kernels with bipartite matching. Inspired by DETR’s [2] reformulation of object detection with proposals based on queries, MaskFormer [11] used transformer-based architecture as a mask classifier. Mask2Former [10] improved upon MaskFormer with learnable queries, deformable multi-scale attention [51] in the decoder, a masked cross-attention and set the new state of the art on all three

tasks. Unfortunately, it requires training the model individually on each task to achieve the best performance. Therefore, there remains a gap in truly unifying the three segmentation tasks. To the best of our knowledge, OneFormer is the first framework to beat state of the art on all three image segmentation tasks with a single universal model.

2.3. Transformer-based Architectures

Architectures based on the transformer encoder-decoder structure [2, 25, 28, 51] have proved effective in object detection since the introduction of DETR [2]. Mask2Former [10, 11] demonstrated the effectiveness of such architectures for image segmentation with a mask classification formulation. Inspired by this success, we also formulate our framework as a query-based mask classification task. Additionally, we claim that calculating a query-text contrastive loss [33, 43] on the task-guided queries can help the model learn inter-task differences and reduce the category mispredictions in the model outputs. Concurrent to our work, LMSeg [50] uses text derived from multiple datasets’ taxonomy to calculate a query-text contrastive loss and tackle the multi-dataset segmentation training challenge. Unlike LMSeg [50], our work focuses on multiple tasks and uses the classes in the training sample’s GT label to calculate the contrastive loss.

3. Method

In this section, we introduce OneFormer, a universal image segmentation framework jointly trained on the panoptic, semantic, and instance segmentation and outperforms individually trained models. We provide an overview of OneFormer in Fig. 2. OneFormer uses two inputs: sample image and task input of the form “the task is {task}”. During our single joint training process, the task is uniformly

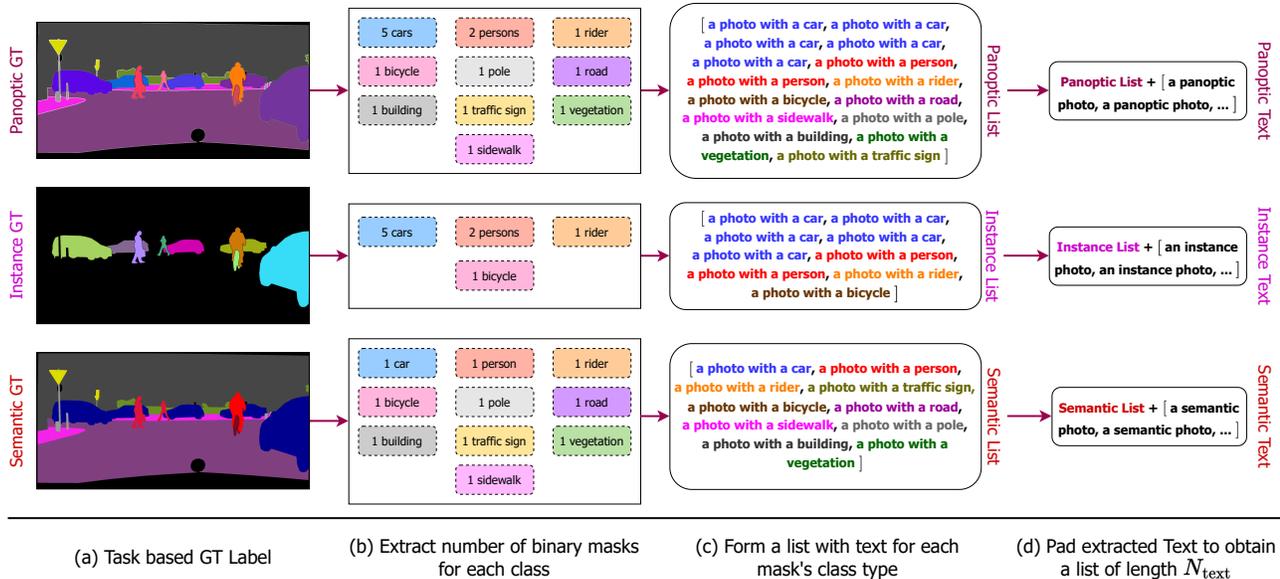


Figure 3. **Input Text Formation.** (a) We uniformly sample the task during training. (b) We extract the number of distinct binary masks for each class from the corresponding GT label. (c) We form a list with text descriptions for each mask using the template “a photo with a {CLS}”, where CLS represents the corresponding class name for the object mask. (d) Finally, we pad the text list to a constant length of N_{text} using “a/an {task} photo” entries which represent the no-object detections; where $\text{task} \in \{\text{panoptic}, \text{instance}, \text{semantic}\}$.

sampled from {panoptic, instance, semantic} for each image. Firstly, we extract multi-scale features from the input image using a backbone and a pixel decoder. We tokenize the task input to obtain a 1-D task token used to condition the object queries and, consequently, our model on the task for each input. Additionally, we create a text list representing the number of binary masks for each class present in the GT label and map it to text query representations. Note that the text list depends on the input image and the {task}. For supervision of the model’s task-dynamic predictions, we derive the corresponding ground-truths from panoptic annotations. As the ground truth is task-dependent, we calculate a query-text contrastive loss between the object and text queries to ensure there is task distinction in the object queries. The object queries and multi-scale features are fed into a transformer decoder to produce final predictions. We provide more details in the following sections.

3.1. Task Conditioned Joint Training

Existing semi-universal architectures for image segmentation [10, 11, 47] face a significant drop in performance when jointly trained on all three segmentation tasks (Tab. 7). We attribute their failure to tackle the multi-task challenge to the absence of task-conditioning in their architecture.

We tackle the multi-task train-once challenge for image segmentation using a task-conditioned joint training strategy. Particularly, we first uniformly sample the task from {panoptic, semantic, instance} for the GT label. We realize the unification potential of panoptic annotations [23] by deriving the task-specific labels from the panoptic annotations,

thus, using only one set of annotations.

Next, we extract a set of binary masks for each category present in the image from the task-specific GT label, *i.e.*, semantic task guarantees only one amorphous binary mask for each class present in the image, whereas, instance task signifies non-overlapping binary masks for only thing classes, ignoring the stuff regions. Panoptic task denotes a single amorphous mask for stuff classes and non-overlapping masks for thing classes as shown in Fig. 3. Subsequently, we iterate over the set of masks to create a list of text (\mathbf{T}_{list}) with a template “a photo with a {CLS}”, where CLS is the class name for the corresponding binary mask. The number of binary masks per sample varies over the dataset. Therefore, we pad \mathbf{T}_{list} with “a/an {task} photo” entries to obtain a padded list (\mathbf{T}_{pad}) of constant length N_{text} , with padded entries representing no-object masks. We later use \mathbf{T}_{pad} for computing a query-text contrastive loss (Sec. 3.3).

We condition our architecture on the task using a task input (\mathbf{I}_{task}) with the template “the task is {task}”, which is tokenized and mapped to a task-token (\mathbf{Q}_{task}). We use \mathbf{Q}_{task} to condition OneFormer on the task (Sec. 3.2).

3.2. Query Representations

During training, we use two sets of queries in our architecture: text queries (\mathbf{Q}_{text}) and object queries (\mathbf{Q}). \mathbf{Q}_{text} is the text-based representation for the segments in the image, while \mathbf{Q} is the image-based representation.

To obtain \mathbf{Q}_{text} , we first tokenize the text entries \mathbf{T}_{pad} and pass the tokenized representations through a text-encoder [43], which is a 6-layer transformer [37]. The en-

coded N_{text} text embeddings represent the number of binary masks and their corresponding classes in the input image. We further concatenate a set of N_{ctx} learnable text context embeddings (\mathbf{Q}_{ctx}) to the encoded text embeddings to obtain the final N text queries (\mathbf{Q}_{text}), as shown in Fig. 4. Our motivation behind using \mathbf{Q}_{ctx} is to learn a unified textual context [48, 49] for a sample image. We only use the text queries during training; therefore, we can drop the text mapper module during inference to reduce the model size.

To obtain \mathbf{Q} , we first initialize the object queries (\mathbf{Q}') as a $N - 1$ times repetitions of the task-token (\mathbf{Q}_{task}). Then, we update \mathbf{Q}' with guidance from the flattened 1/4-scale features inside a 2-layer transformer [2, 37]. The updated \mathbf{Q}' from the transformer (rich with image-contextual information) is concatenated with \mathbf{Q}_{task} to obtain a task-conditioned representation of N queries, \mathbf{Q} . Unlike the vanilla all-zeros or random initialization [2], the task-guided initialization of the queries and the concatenation with \mathbf{Q}_{task} is critical for the model to learn multiple segmentation tasks (Sec. 4.3).

3.3. Task Guided Contrastive Queries

Developing a single model for all three segmentation tasks is challenging due to the inherent differences among the three tasks. The meaning of the object queries, \mathbf{Q} , is task-dependent. Should the queries focus only on the thing classes (instance segmentation), or should the queries predict only one amorphous object for each class present in the image (semantic segmentation) or a mix of both (panoptic segmentation)? Existing query-based architectures [10, 11] do not take such differences into account and hence, fail at effectively training a single model on all three tasks.

To this end, we propose calculating a query-text contrastive loss using \mathbf{Q} and \mathbf{Q}_{text} . We use \mathbf{T}_{pad} to obtain the text queries representation, \mathbf{Q}_{text} , where \mathbf{T}_{pad} is a list of textual representations for each mask-to-be-detected in a given image with “a/an {task} photo” representing the no-object detections in \mathbf{Q} [2]. Thus, the text queries align with the purpose of object queries, representing the objects/segments present [2] in an image. Therefore, we can successfully learn the inter-task distinctions in the query representations using a contrastive loss between the ground truth-derived text and object queries. Moreover, contrastive learning on the queries enables us to attend to inter-class differences and reduce category misclassifications.

$$\begin{aligned} \mathcal{L}_{\mathbf{Q} \rightarrow \mathbf{Q}_{\text{text}}} &= -\frac{1}{B} \sum_{i=1}^B \log \frac{\exp(q_i^{\text{obj}} \odot q_i^{\text{txt}} / \tau)}{\sum_{j=1}^B \exp(q_i^{\text{obj}} \odot q_j^{\text{txt}} / \tau)}, \\ \mathcal{L}_{\mathbf{Q}_{\text{text}} \rightarrow \mathbf{Q}} &= -\frac{1}{B} \sum_{i=1}^B \log \frac{\exp(q_i^{\text{txt}} \odot q_i^{\text{obj}} / \tau)}{\sum_{j=1}^B \exp(q_i^{\text{txt}} \odot q_j^{\text{obj}} / \tau)} \quad (1) \\ \mathcal{L}_{\mathbf{Q} \leftrightarrow \mathbf{Q}_{\text{text}}} &= \mathcal{L}_{\mathbf{Q} \rightarrow \mathbf{Q}_{\text{text}}} + \mathcal{L}_{\mathbf{Q}_{\text{text}} \rightarrow \mathbf{Q}} \end{aligned}$$

Considering that we have a batch of B object-text query pairs $\{(q_i^{\text{obj}}, x_i^{\text{txt}})\}_{i=1}^B$, where q_i^{obj} and q_i^{txt} are the correspond-

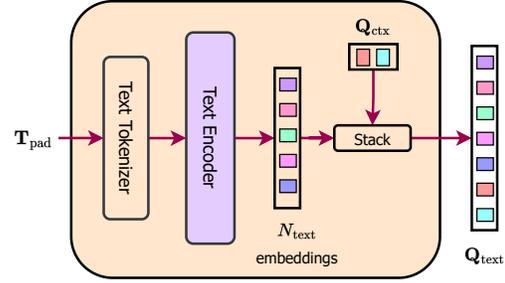


Figure 4. **Text Mapper.** We tokenize and then encode the input text list (\mathbf{T}_{pad}) using a 6-layer transformer text encoder [37, 43] to obtain a set of N_{text} embeddings. We concatenate a set of N_{ctx} learnable embeddings to the encoded representations to obtain the final N text queries (\mathbf{Q}_{text}). The N text queries stand for a text-based representation of the objects present in an image.

ing object and text queries, respectively, of the i -th pair. We measure the similarity between the queries by calculating a dot product. The total contrastive loss is composed of [43]: (i) an object-to-text ($\mathcal{L}_{\mathbf{Q} \rightarrow \mathbf{Q}_{\text{text}}}$) and; (ii) a text-to-object contrastive loss ($\mathcal{L}_{\mathbf{Q}_{\text{text}} \rightarrow \mathbf{Q}}$) as shown in Eq. (1). τ is a learnable temperature parameter to scale the contrastive logits.

3.4. Other Architecture Components

Backbone and Pixel Decoder: We use the widely used ImageNet [24] pre-trained backbones [17, 30, 31] to extract multi-scale feature representations from the input image. Our pixel decoder aids the feature modeling by gradually upsampling the backbone features. Motivated by the recent success of multi-scale deformable attention [10, 51], we use the same Multi-Scale Deformable Transformer (MSDeformAttn) based architecture for our pixel decoder.

Transformer Decoder: We use a multi-scale strategy [10] to utilize the higher resolution maps inside our transformer decoder. Specifically, we feed the object queries (\mathbf{Q}) and the multi-scale outputs from the pixel decoder (F_i), $i \in \{1/4, 1/8, 1/16, 1/32\}$ as inputs. We use the features with resolution 1/8, 1/16 and 1/32 of the original image alternately to update \mathbf{Q} using a masked cross-attention (CA) operation [10], followed by a self-attention (SA) and finally a feed-forward network (FFN). We perform these sets of alternate operations L times inside the transformer decoder.

The final query outputs from the transformer decoder are mapped to a $K + 1$ dimensional space for class predictions, where K denotes the number of classes and an extra $+1$ for the no-object predictions. To obtain the final masks, we decode $F_{1/4}$ with the help of an einsum operation between \mathbf{Q} and $F_{1/4}$. During inference, we follow the same post-processing technique as [10] to obtain the final panoptic, semantic, and instance segmentation predictions. We keep predictions with scores above a threshold of 0.5, 0.8, and 0.8 during panoptic post-processing on the ADE20K [13], Cityscapes [12] and COCO [27] datasets, respectively.

Method	Backbone	#Params	#FLOPs	#Queries	Crop Size	Iters	PQ	AP	mIoU (s.s.)	mIoU (m.s.)
<i>Individual Training</i>										
UPerNet [‡] [41]	SwinV2-L [†] [29]	—	—	—	640×640	40k	—	—	—	55.9
SeMask Mask2Former [21]	SeMask Swin-L [†] [21]	223M	426G	200	640×640	160k	—	—	56.4	57.5
UPerNet + K-Net [47]	Swin-L [†] [30]	—	—	—	640×640	160k	—	—	—	54.3
MaskFormer [11]	Swin-L [†] [30]	212M	375G	100	640×640	160k	—	—	54.1	55.6
Mask2Former-Panoptic* [10]	Swin-L [†] [30]	216M	413G	200	640×640	160k	48.7	34.2	54.5	—
Mask2Former-Instance [10]	Swin-L [†] [30]	216M	411G	200	640×640	160k	—	34.9	—	—
Mask2Former-Semantic [10]	Swin-L [†] [30]	215M	403G	100	640×640	160k	—	—	56.1	57.3
UPerNet ^{‡‡} [41]	SwinV2-G ^{†‡} [29]	>3B	—	—	640×640	80k	—	—	59.1	—
Mask2Former ^{‡‡} [10]	BEiT-3 ^{†‡} [40]	1.9B	—	—	896×896	—	—	—	62.0	62.8
<i>Joint Training</i>										
OneFormer	Swin-L [†] [30]	219M	436G	250	640×640	160k	49.8	35.9	57.0	57.7
OneFormer	Swin-L [†] [30]	219M	801G	250	896×896	160k	51.1	37.6	57.4	58.3
OneFormer	ConvNeXt-L [†] [31]	220M	389G	250	640×640	160k	50.0	36.2	56.6	57.4
OneFormer	ConvNeXt-XL [†] [31]	372M	607G	250	640×640	160k	50.1	36.3	57.4	58.8
OneFormer	DiNAT-L [†] [17]	223M	359G	250	640×640	160k	50.5	36.0	58.3	58.4
OneFormer	DiNAT-L [†] [17]	223M	678G	250	896×896	160k	51.2	36.8	58.1	58.6

Table 1. **SOTA Comparison on the ADE20K val set.** †: backbones pretrained on ImageNet-22K, ‡: trained with batch size 32; *: 0.5 confidence threshold; ‡‡: batch size 64. OneFormer outperforms the individually trained Mask2Former [10]. Mask2Former’s performance with 250 queries is not listed, as its performance degrades with 250 queries. We compute FLOPs using the corresponding crop size.

3.5. Losses

In addition to the contrastive loss on the queries, we calculate the standard classification CE-loss (\mathcal{L}_{cls}) over the class predictions. Following [10], we use a combination of binary cross-entropy (\mathcal{L}_{bce}) and dice loss (\mathcal{L}_{dice}) over the mask predictions. Therefore, our final loss function is a weighted sum of the four losses (Eq. (2)). We empirically set $\lambda_{Q \leftrightarrow Q_{text}} = 0.5$, $\lambda_{cls} = 2$, $\lambda_{bce} = 5$ and $\lambda_{dice} = 5$. To find the least cost assignment, we use bipartite matching [2, 11] between the set predictions and the ground truths. We set λ_{cls} as 0.1 for the no-object predictions [10].

$$\mathcal{L}_{final} = \lambda_{Q \leftrightarrow Q_{text}} \mathcal{L}_{Q \leftrightarrow Q_{text}} + \lambda_{cls} \mathcal{L}_{cls} + \lambda_{bce} \mathcal{L}_{bce} + \lambda_{dice} \mathcal{L}_{dice} \quad (2)$$

4. Experiments

We illustrate that OneFormer, when trained only once with our task-conditioned joint-training strategy, generalizes well to all three image segmentation tasks on three widely used datasets. Furthermore, we provide extensive ablations to demonstrate the significance of OneFormer’s components. Due to space constraints, we provide implementation details in the appendix.

4.1. Datasets and Evaluation Metrics

Datasets. We experiment on three widely used datasets that support all three: semantic, instance, and panoptic segmentation tasks. **Cityscapes** [12] consists of a total 19 (11 “stuff” and 8 “thing”) classes with 2,975 training, 500 validation and 1,525 test images. **ADE20K** [13] is another benchmark dataset with 150 (50 “stuff” and 100 “thing”) classes among the 20,210 training and 2,000 validation images. **COCO** [27] has 133 (53 “stuff” and 80 “thing”) classes with 118k training and 5,000 validation images.

Evaluation Metrics. For all three image segmentation tasks, we report the **PQ** [23], **AP** [27], and **mIoU** [15] scores. Since we only have a single model for all three tasks, we use the value of the task token to decide the scores to consider. For *e.g.*, when task is panoptic, we report the **PQ** score and similarly we report **AP** and **mIoU** scores when task is instance and semantic, respectively.

4.2. Main Results

ADE20K. We compare OneFormer with the existing state-of-the-art pseudo-universal and specialized architectures on the ADE20K [13] val dataset in Tab. 1. With the standard Swin-L[†] backbone, OneFormer, while being trained only once, outperforms Mask2Former’s [10] individually trained models on all three image segmentation tasks and sets a new state-of-the-art performance when compared with other methods using the same backbone.

Cityscapes. We compare OneFormer with the existing state-of-the-art pseudo-universal and specialized architectures on the Cityscapes [13] val dataset in Tab. 2. With Swin-L[†] backbone, OneFormer outperforms Mask2Former with a +0.6% and +1.9% improvement on the **PQ** and **AP** metrics, respectively. Additionally, with ConvNeXt-L[†] and ConvNeXt-XL[†] backbone, OneFormer sets a new state-of-the-art of 68.5% PQ and 46.7% AP, respectively.

COCO. We compare OneFormer with the existing state-of-the-art pseudo-universal and specialized architectures on the COCO [27] val2017 dataset in Tab. 3. With Swin-L[†] backbone, OneFormer performs on-par with the individually trained Mask2Former [10] with a +0.1% improvement in the PQ score. Due to the discrepancies between the panoptic and instance annotations in COCO [27], we evaluate the AP score using the instance ground truths derived

Method	Backbone	#Params	#FLOPs	#Queries	Crop Size	Iters	PQ	AP	mIoU (s.s.)	mIoU (m.s.)
<i>Individual Training</i>										
CMT-DeepLab [‡] [45]	MaX-S [†] [38]	—	—	—	1025×2049	60k	64.6	—	81.4	—
Axial-DeepLab-L [‡] [39]	Axial ResNet-L [†] [39]	45M	687G	—	1025×2049	60k	63.9	35.8	81.0	81.5
Axial-DeepLab-XL [‡] [39]	Axial ResNet-XL [†] [39]	173M	2447G	—	1025×2049	60k	64.4	36.7	80.6	81.1
Panoptic-DeepLab [‡] [9]	SWideRNet [†] [7]	536M	10365G	—	1025×2049	60k	66.4	40.1	82.2	82.9
Mask2Former-Panoptic [10]	Swin-L [†] [30]	216M	514G	200	512×1024	90k	66.6	43.6	82.9	—
Mask2Former-Instance [10]	Swin-L [†] [30]	216M	507G	200	512×1024	90k	—	43.7	—	—
Mask2Former-Semantic [10]	Swin-L [†] [30]	215M	494G	100	512×1024	90k	—	—	83.3	84.3
kMaX-DeepLab [‡] [46]	ConvNeXt-L [†] [31]	232M	1673G	256	1025×2049	60k	68.4	44.0	83.5	—
<i>Joint Training</i>										
OneFormer	Swin-L [†] [30]	219M	543G	250	512×1024	90k	67.2	45.6	83.0	84.4
OneFormer	ConvNeXt-L [†] [31]	220M	497G	250	512×1024	90k	68.5	46.5	83.0	84.0
OneFormer	ConvNeXt-XL [†] [31]	372M	775G	250	512×1024	90k	68.4	46.7	83.6	84.6
OneFormer	DiNAT-L [†] [17]	223M	450G	250	512×1024	90k	67.6	45.6	83.1	84.0

Table 2. **SOTA Comparison on Cityscapes val set.** †: backbones pretrained on ImageNet-22K; ‡: trained with batch size 32, *: hidden dimension 1024. OneFormer outperforms the individually trained Mask2Former [10] models. Mask2Former’s performance with 250 queries is not listed, as its performance degrades with 250 queries. We compute FLOPs using the corresponding crop size.

Method	Backbone	#Params	#FLOPs	#Queries	Epochs	PQ	PQ Th	PQ St	AP	AP ^{Instance}	mIoU
<i>Individual Training</i>											
MaskFormer [11]	Swin-L [†] [30]	212M	792G	100	300	52.7	58.5	44.0	—	—	64.8
K-Net [47]	Swin-L [†] [30]	—	—	100	36	54.6	60.2	46.0	—	—	—
Panoptic SegFormer [26]	Swin-L [†] [30]	221M	816G	353	24	55.8	61.7	46.9	—	—	—
Mask2Former-Panoptic [10]	Swin-L [†] [30]	216M	875G	200	100	57.8	64.2	48.1	48.7	48.6	67.4
Mask2Former-Instance [10]	Swin-L [†] [30]	216M	868G	200	100	—	—	—	49.1	50.1	—
Mask2Former-Semantic [‡] [10]	Swin-L [†] [30]	216M	891G	200	100	—	—	—	—	—	67.2
kMaX-DeepLab* [46]	ConvNeXt-L [†] [31]	232M	749G	128	81	57.9	64.0	48.6	—	—	—
kMaX-DeepLab* [46]	ConvNeXt-L [†] [31]	232M	749G	256	81	58.0	64.2	48.6	—	—	—
<i>Joint Training</i>											
OneFormer	Swin-L [†] [30]	219M	891G	150	100	57.9	64.4	48.0	49.0	48.9	67.4
OneFormer	DiNAT-L [†] [17]	223M	736G	150	100	58.0	64.3	48.4	49.2	49.2	68.1

Table 3. **SOTA Comparison on COCO val2017 set.** †: Imagenet-22k pretrained; ‡: retrained model; *: trained with batch size 64. OneFormer competes with the individually trained Mask2Former [10]. We evaluate the AP score on instance ground truths derived from the panoptic annotations. Mask2Former’s performance with 150 queries is not listed, as its performance degrades with 150 queries. We compute FLOPs using 100 validation COCO images (varying sizes). AP^{Instance} represents evaluation on the original instance annotations.

from the panoptic annotations. We provide more information in the appendix. Following [10], we evaluate mIoU on semantic ground truths derived from panoptic annotations.

4.3. Ablation Studies

We analyze OneFormer’s components through a series of ablation studies. Unless stated otherwise, we ablate with Swin-L[†] OneFormer on the Cityscapes [12] dataset.

Task-Conditioned Architecture. We validate the importance of the task token (\mathbf{Q}_{task}), initializing the queries with repetitions of the task token (task-guided query init.) and the learnable text context (\mathbf{Q}_{ctx}) by removing each component one at a time in Tab. 4. Without the task token, we observe a significant drop in the AP score (−2.7%). Furthermore, using a learnable text context (\mathbf{Q}_{ctx}) leads to an improvement of +4.5% in the PQ score, proving its significance. Lastly, initializing the queries as repetitions of the task token (task-guided query init.) instead of using an all-zeroes initialization [2] leads to an improvement of +1.4%

in the PQ and +1.1% in the AP score, indicating the importance of task-conditioning the initialization of the queries.

Contrastive Query Loss. We report results without the query-text contrastive loss ($\mathcal{L}_{\mathbf{Q} \leftrightarrow \mathbf{Q}_{\text{ext}}}$) in Tab. 5. We observe that the contrastive loss significantly benefits the PQ (+8.4%) and AP (+3.2%) scores. We also conduct experiments substituting our query-text contrastive loss with a classification loss (\mathcal{L}_{cls}) on the queries. \mathcal{L}_{cls} can be regarded as a straightforward alternative for $\mathcal{L}_{\mathbf{Q} \leftrightarrow \mathbf{Q}_{\text{ext}}}$ as both provide supervision for the number of masks for each class present in the image. However, we observe significant drops on all the metrics (−0.8% PQ, −0.9% AP, and −0.4% mIoU) using the classification loss instead of the contrastive loss. We attribute the drops to the inability of the classification loss to capture the inter-task differences effectively.

Input Text Template. We study the importance of the template choice for the entries in the text list (\mathbf{T}_{list}) in Tab. 6. We experiment with “a photo with a {CLS} {TYPE}” template for our text entries where CLS is the class name for the object

	PQ	AP	mIoU
OneFormer (ours)	67.2	45.6	83.0
– task-token (\mathbf{Q}_{task})	66.5 (-0.7)	43.3 (-2.3)	82.9 (-0.1)
– learnable text context (\mathbf{Q}_{ctx})	62.7 (-4.5)	45.0 (-0.6)	82.8 (-0.2)
– task-guided query init.	65.8 (-1.4)	44.5 (-1.1)	83.1 (+0.1)

Table 4. **Ablation on Components.** A task-conditioned architecture significantly improves the AP scores and using learnable text context improves the PQ score.

	PQ	AP	mIoU.	#param.
contrastive-loss (ours)	67.2	45.6	83.0	219M
query classification-loss	66.4 (-0.8)	44.7 (-0.9)	82.6 (-0.4)	219M
no contrastive-loss	58.8 (-8.4)	42.4 (-3.2)	82.5 (-0.5)	219M

Table 5. **Ablation on Loss.** The contrastive loss is essential for learning the inter-task distinctions during training.

	PQ	AP	mIoU
“a photo with a {CLS}” (ours)	67.2	45.6	83.0
“a photo with a {CLS} {TYPE}”	65.4 (-1.8)	44.5 (-1.1)	82.8 (-0.2)
“{CLS}”	66.6 (-0.6)	44.7 (-0.9)	82.5 (-0.5)

Table 6. **Ablation on Input Text Templates.** The template for the input text list entries is a critical factor for good performance. CLS represents the class name and TYPE stands for the stuff/thing.

mask and TYPE is the task-dependent class-type: “stuff” for amorphous masks (panoptic and semantic task) and “thing” for all distinct object masks. We also experiment with the identity template “{CLS}”. Our choice of the template: “a photo with a {CLS}” gives a strong performance as a baseline. We believe more exploration in the text template space could help in improving the performance further.

Task Conditioned Joint Training. We train a baseline Swin-L[†] Mask2Former-Joint model with our joint training strategy on the ADE20K [13] dataset. We compare the Mask2Former-Joint baseline with our Swin-L[†] OneFormer in Tab. 7. We train both models for 160k iterations with a batch size of 16. Our OneFormer achieves a +2.3%, +2.2%, and +0.8% improvement on the PQ, AP and mIoU metrics, respectively, proving the importance of our architecture design for practical multi-task joint training.

Task Token Input. We demonstrate that our framework is sensitive to the task token input by setting the value of {task} during inference as panoptic, instance, or semantic in Tab. 8. We report results with our Swin-L[†] OneFormer trained on ADE20K [13] dataset. We observe a significant drop in the PQ and mIoU metrics when task is instance compared to panoptic. Moreover, the PQSt drops to 1.5%, and there is only a -0.8% drop on PQTh metric, proving that the network learns to focus majorly on the distinct “thing” instances when the task is instance. Similarly, there is a sizable drop in the PQ, PQTh and AP metrics for the semantic task with PQSt staying the same, showing that our framework can segment out amorphous masks for “stuff” regions but does not predict different masks for “thing” objects. Therefore, OneFormer dynamically learns the inter-task distinctions, which is critical for a train-once multi-task

	PQ	AP	mIoU	#param.
OneFormer (ours)	49.8	35.9	57.0	219M
Mask2Former-Joint	48.7 (-1.1)	33.7 (-2.2)	56.2 (-0.8)	216M

Table 7. **Ablation on Joint Training.** OneFormer significantly beats the baseline’s scores. We report results with Swin-L[†] [30] backbone trained for 160k iterations on the ADE20K [13] dataset.

Task Token Input	PQ	PQ Th	PQ St	AP	mIoU
the task is panoptic	49.3	49.6	50.2	35.8	57.0
the task is instance	33.1	48.8	1.5	35.9	26.4
the task is semantic	40.4	35.5	50.2	25.3	57.0

Table 8. **Ablation on Task Token Input.** Our OneFormer is sensitive to the input task token value. We report results with Swin-L[†] OneFormer on the ADE20K [13] val set. The numbers in pink denote results on secondary task metrics.

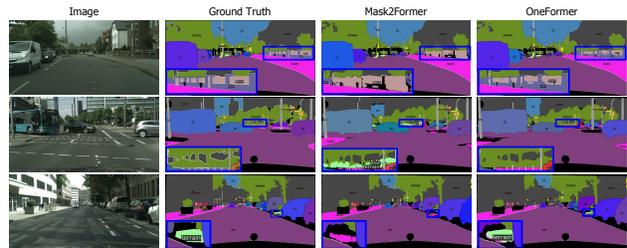


Figure 5. **Reduced Category Misclassifications.** Our OneFormer segments the regions (inside blue boxes) with similar classes more accurately than Mask2Former [10]. **Zoom in for best view.**

architecture. We include qualitative analysis on the task dynamic nature of OneFormer in the appendix.

Reduced Category Misclassifications. Our query-text contrastive loss helps OneFormer learn the inter-task distinctions and reduce the number of category misclassifications in the predictions. Mask2Former incorrectly predicts “wall” as “fence” in the first row, “vegetation” as “terrain”, and “terrain” as “sidewalk”. At the same time, our OneFormer produces more accurate predictions in regions (inside blue boxes) with similar classes, as shown in Fig. 5.

5. Conclusion

We present OneFormer, a transformer-based multi-task universal image segmentation framework with task-guided queries to unify the three image segmentation tasks with a single universal architecture, a single model, and training on a single dataset. Our jointly trained single OneFormer model outperforms the individually trained specialized Mask2Former models, the previous single-architecture state of the art, on all three segmentation tasks across major datasets. Consequently, OneFormer can reduce training time, weight storage, and inference hosting requirements to a third. We believe OneFormer is a significant step towards making image segmentation more universal and accessible.

Acknowledgments. We thank the U of Oregon, UIUC, Picsart AI Research, and IARPA (Contract No. 2022-21102100004) for generously supporting this work.

References

- [1] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: Delving into high quality object detection. In *CVPR*, 2018. 3
- [2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2, 3, 5, 6, 7
- [3] Joao Carreira, Rui Caseiro, Jorge Batista, and Cristian Sminchisescu. Semantic segmentation with second-order pooling. In *ECCV*, 2012. 2
- [4] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Hybrid task cascade for instance segmentation. In *CVPR*, 2019. 3
- [5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 2
- [6] 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. In *TPAMI*, 2017. 1, 2
- [7] Liang-Chieh Chen, Huiyu Wang, and Siyuan Qiao. Scaling wide residual networks for panoptic segmentation. *arXiv*, 2020. 7
- [8] Bowen Cheng, Liang-Chieh Chen, Yunchao Wei, Yukun Zhu, Zilong Huang, Jinjun Xiong, Thomas S Huang, Wen-Mei Hwu, and Honghui Shi. Spgnet: Semantic prediction guidance for scene parsing. In *CVPR*, 2019. 2
- [9] 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, 7
- [10] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 1, 2, 3, 4, 5, 6, 7, 8
- [11] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *NeurIPS*, 2021. 2, 3, 4, 5, 6, 7
- [12] 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. 2, 5, 6, 7
- [13] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. Semantic understanding of scenes through the ade20k dataset. In *CVPR*, 2017. 2, 5, 6, 8
- [14] Jifeng Dai, Kaiming He, and Jian Sun. Convolutional feature masking for joint object and stuff segmentation. In *CVPR*, 2015. 2
- [15] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes challenge: A retrospective. *IJCV*, 2015. 6
- [16] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *ECCV*, 2014. 2
- [17] Ali Hassani and Humphrey Shi. Dilated neighborhood attention transformer. *arXiv:2209.15001*, 2022. 2, 5, 6, 7
- [18] Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood attention transformer. *arXiv:2204.07143*, 2022. 2
- [19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 1, 3
- [20] Zilong Huang, Xinggang Wang, Yunchao Wei, Lichao Huang, Humphrey Shi, Wenyu Liu, and Thomas S. Huang. Ccnet: Criss-cross attention for semantic segmentation. In *TPAMI*, 2020. 2
- [21] Jitesh Jain, Anukriti Singh, Nikita Orlov, Zilong Huang, Jiachen Li, Steven Walton, and Humphrey Shi. Semask: Semantically masking transformer backbones for effective semantic segmentation. *arXiv*, 2021. 2, 6
- [22] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, 2019. 3
- [23] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, 2019. 1, 2, 3, 4, 6
- [24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012. 5
- [25] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *CVPR*, pages 13619–13627, 2022. 3
- [26] Zhiqi Li, Wenhai Wang, Enze Xie, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, Tong Lu, and Ping Luo. Panoptic segformer: Delving deeper into panoptic segmentation with transformers. In *CVPR*, 2022. 7

- [27] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context. In *ECCV*, 2014. 2, 5, 6
- [28] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. DAB-DETR: Dynamic anchor boxes are better queries for DETR. In *ICLR*, 2022. 3
- [29] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. *arXiv*, 2021. 6
- [30] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 2, 5, 6, 7, 8
- [31] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *CVPR*, 2022. 2, 5, 6, 7
- [32] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1, 2
- [33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *arXiv*, 2021. 2, 3
- [34] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *ICCV*, 2021. 2
- [35] Joseph Tighe, Marc Niethammer, and Svetlana Lazebnik. Scene parsing with object instances and occlusion ordering. In *CVPR*, 2014. 3
- [36] Z. Tu, Xiangrong Chen, Alan Yuille, and Song Zhu. Image parsing: Unifying segmentation, detection, and recognition. In *IJCV*, 2005. 3
- [37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 3, 4, 5
- [38] 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. 3, 7
- [39] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-DeepLab: Stand-alone axial-attention for panoptic segmentation. In *ECCV*, 2020. 3, 7
- [40] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, and Furu Wei. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv*, 2022. 6
- [41] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, 2018. 6
- [42] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *NeurIPS*, 2021. 2
- [43] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *CVPR*, 2022. 2, 3, 4, 5
- [44] Jian Yao, Sanja Fidler, and Raquel Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *CVPR*, 2012. 3
- [45] Qihang Yu, Huiyu Wang, Dahun Kim, Siyuan Qiao, Maxwell Collins, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Cmt-deeplab: Clustering mask transformers for panoptic segmentation. In *CVPR*, 2022. 3, 7
- [46] Qihang Yu, Huiyu Wang, Siyuan Qiao, Maxwell Collins, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. k-means mask transformer. In *ECCV*, 2022. 2, 3, 7
- [47] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-Net: Towards unified image segmentation. In *NeurIPS*, 2021. 1, 2, 3, 4, 6, 7
- [48] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 5
- [49] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision (IJCV)*, 2022. 5
- [50] Qiang Zhou, Yuang Liu, Chaohui Yu, Jingliang Li, Zhibin Wang, and Fan Wang. LMSeg: Language-guided multi-dataset segmentation. In *ICLR*, 2023. 3
- [51] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv*, 2020. 3, 5