A Bottom-Up Approach to Class-Agnostic Image Segmentation

Sebastian Dille[1], Ari Blondal[1,2], Sylvain Paris[3], and Yağız Aksoy[1]

[1] Simon Fraser University, BC, Canada
[2] McGill University, QC, Canada
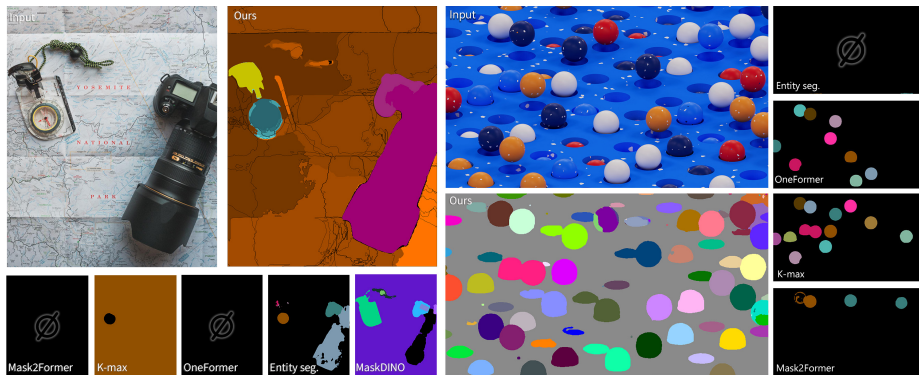[3] Adobe Research, MA, United States

Fig. 1: We introduce a bottom-up approach to class-agnostic image segmentation. We show that our formulation leads to generalization to images in-the-wild that are not well-represented in common training datasets. We generate detailed segmentation maps for complex scenes where other class-based or class-agnostic approaches fall short.

Abstract. Class-agnostic image segmentation is a crucial component in automating image editing workflows, especially in contexts where object selection traditionally involves interactive tools. Existing methods in the literature often adhere to top-down formulations, following the paradigm of class-based approaches, where object detection precedes per-object segmentation. In this work, we present a novel bottom-up formulation for addressing the class-agnostic segmentation problem. We supervise our network directly on the projective sphere of its feature space, employing losses inspired by metric learning literature as well as losses defined in a novel segmentation-space representation. The segmentation results are obtained through a straightforward mean-shift clustering of the estimated features. Our bottom-up formulation exhibits exceptional generalization capability, even when trained on datasets designed for class-based segmentation. We further showcase the effectiveness of our generic approach by addressing the challenging task of cell and nucleus segmentation. We believe that our bottom-up formulation will offer valuable insights into diverse segmentation challenges in the literature.

Keywords: Image Segmentation · Pixel Clustering · Metric Learning · Hyperspherical Learning · Contrastive Learning

1   Introduction

In most image editing scenarios, object selection is the first step for localized image editing or compositing. Automating the object selection, hence, is an interesting application scenario for increased productivity. A generalized object selection method requires the segmentation of every object in any image in the wild. This is a challenging task that standard class-based image segmentation approaches such as semantic or panoptic segmentation fail to accomplish due to the inherently limited number of classes labeled in a training dataset.

With this motivation, recent literature focuses on the *class-agnostic* segmentation problem. Open-set panoptic segmentation [25,63] approaches this problem by extending the label space in panoptic segmentation with an *unknown* class, aiming to detect objects that do not fit in the set of defined classes. Open-world entity segmentation [51, 52], on the other hand, defines the segmentation problem as fully class-agnostic, and presents a method that can detect object centers, which are then used to create the final segmentation map. Segment anything [33] adopts a prompt-based approach and conducts class-agnostic segmentation by assuming a regular grid of user inputs, powered by an extensive dataset.

All these approaches follow a *top-down* approach to segmentation, where the first task of the system is to detect the objects in the scene, followed by per-object segmentation. This top-down approach is in contrast with our understanding of human cognition. Humans can easily identify objects or coherent regions in a wide variety of realistic or abstract, complex or simple images. The dominant process in human object detection is modeled to be *bottom-up* [4], where grouping of features in the scene is followed by object detection and finally classification.

In this work, we present a novel approach to class-agnostic image segmentation with a bottom-up formulation. We adopt the entity definition by Qi *et al.* [52] that unifies *things* and *stuff* into classless entities. We develop our formulation in a feature space with projective geometry, generating per-pixel features that are parallel to each other within the same entity, and orthogonal to all features outside their entity. This allows for maximally separated entities that can conveniently be clustered with simple mean-shift clustering for a dense class-agnostic segmentation during inference. We achieve this with a loss combination inspired by metric learning and a novel segmentation-space formulation that allows for the backpropagation of segmentation-focused losses into our hyper-dimensional feature space.

Our formulation is carefully developed for generalization to class-agnostic only through class-based datasets. Despite using the standard segmentation datasets MS COCO [5, 40], ADE-20k [66], and CIHP [21] as our only real-world training data, our bottom-up approach shows an exceptional generalization ability to unseen classes as well as out-of-distribution images as Figure 1 shows. We demonstrate the performance of our system through zero-shot quantitative analysis. Despite utilizing a smaller architecture, we show that we can generate detailed segmentations for complex scenes in the wild. We further demonstrate the generic nature of our bottom-up formulation by improving upon the state-of-the-art in cell and nucleus segmentation.

2   Related Work

The field of automatic image segmentation is dominated by class-based object labeling approaches [6,9,11,13,14,18,24,31,32,37,43,54,56,59,60]. These methods are trained to recognize objects from a fixed set of known classes and assign pixel labels accordingly. Depending on the application scenario, the algorithms are either identifying semantics alone [9,43,54], distinguishing individual instances of countable objects [6,18,24,56,59,60], or combining both in a panoptic fashion [11, 13,14,31,32,37]. Their inherent inability to generalize to unseen classes, however, makes them less suitable for use in image editing. We focus on class-agnostic segmentation below and refer to the recent survey [47] for an in-depth review.

Class-agnostic Image Segmentation   Recently, a growing number of segmentation approaches [25, 50–52, 63] are removing class dependency to handle out-of-distribution objects and improving generalization: Open-set panoptic segmentation methods [25, 63] on the one hand are still closely following the concept of panoptic segmentation but introduce an additional class to the training set to label unknown elements. Once identified, the corresponding areas are further segmented via class-agnostic clustering based on predicted bounding boxes.

Entity-segmentation methods [50–52] on the other hand entirely remove semantic information from the training process, treating each object in the dataset as a unique entity. Qi *et al.* [52] first introduce this concept by replacing the supervision from a proposal-based segmentation approach [56] with class-agnostic masks. They show how this change alone results in increased generalization capability and further adapt the concept in subsequent work for pretraining in class-based segmentation [50] and to generate high-resolution results [51]. Both works formulate segmentation in a top-down fashion, incorporating a proposal generator to predict bounding boxes [50] or entity centers [51]. This limits their generalization ability to objects that do not match the training distribution in appearance. In contrast, we construct our method as a bottom-up framework based on *object discrimination* that generates segments by clustering on the hypersphere and is independent of the exact appearance.

Segment anything [33] takes inspiration from prompt-based natural language processing approaches and formulates the segmentation problem with various forms of input. For the class-agnostic problem akin to entity segmentation, they assume a regular grid of input prompts to generate their dense output and demonstrate a strong generalization ability enabled by their immense dataset. One major shortcoming of prompt-based methods is the dependence on very large training datasets, which limits their applicability to other domains where collection of such datasets is prohibitively expensive such as commercial applications, medical image segmentation, and fine-grained segmentation.

Our bottom-up approach, on the other hand, is designed to leverage small or incomplete datasets while still achieving generalizability. We achieve similar performance to the segment anything model with similar number of parameters to ours, despite them training on a dataset that is larger than ours by 2 orders of magnitude. We also achieve state-of-the-art performance in cell and nucleus seg-

mentation compared to domain-specific approaches, demonstrating the generic use of our bottom-up formulation in problems with limited training data. Our representation-based metric learning approach can further be integrated into future prompt-based approaches to improve their performance in problems where large-scale data acquisition has inherent challenges.

Entity Representation in Projective Space   Object discrimination is a long-existing concept in image processing in the form of clustering-based methods [15, 28, 44, 49, 57, 64] and is recently being revisited by instance segmentation approaches with the goal to distinguish individual instances within an already recognized "thing"-category [2, 11, 16, 19, 34, 58]. In this constrained setting, the prediction accuracy can be greatly increased by computing affinities on high-dimensional features instead of pixels and applying contrastive losses as class-agnostic supervision. For panoptic segmentation, the bottom-up approach is challenging since the combination of different elements within the same "stuff" ground truth category creates ambiguities, and so far only combined approaches have been proposed with bottom-up "thing" discrimination and top-down segmentation for "stuff" [11]. We argue that a careful definition of the feature representation and supervision space is crucial for bottom-up panoptic segmentation.

We formulate our supervision thus on the projective sphere, a hypersphere with antipodal equivalence. Non-euclidean representations have seen growing attention in grouping tasks due to the inherent hierarchical properties in hyperbolic space [1, 8, 20, 30, 38, 48, 61] and intuitive metric learning on the hypersphere [10, 23, 26, 34, 67]. In line with works by Kong *et al.* [34] and Hwang *et al.* [26], we use cosine similarity to define a metric on the projective sphere. By supervising directly in this feature space, we yield an entity-specific representation that allows us to ignore ambiguous background regions during training and to apply simple mean-shift clustering during inference.

3   Generating Distinguishable Features

We approach the class-agnostic image segmentation problem by looking back at the most basic definition of image segmentation. Our aim is to generate an image representation that allows us to cluster the pixels in the image into segments that correspond to different entities. For this purpose, we formulate our training and inference scheme purely in our projective spherical feature space using losses inspired by contrastive learning. We also define a low-dimensional segmentation space that allows us to signal clustering performance to the network during training. We show that our features can be used to segment the image using a simple mean-shift clustering formulation.

3.1   Feature Representation

We define our feature space as the real projective sphere. This means that it resembles a hyper-dimensional sphere of radius 1, where each feature - encoded through unit homogeneous coordinates - forms a point on the surface, and points

on opposite sides of the sphere are equivalent. Because of the equivalency, these points resemble *lines* through the origin in d-dimensional space, a concept that also better illustrates our representation's focus on angle distance, parallelism, and orthogonality. Our feature space is thus defined as $\mathcal{F}$:

$$\mathcal{F} = \{\boldsymbol{f} \in \mathbb{R}^d : ||\boldsymbol{f}||_2 = 1, \boldsymbol{f} = -\boldsymbol{f}\}, \tag{1}$$

and we endow it with a distance metric defined as the cosine distance between two lines:

$$dist(\boldsymbol{f}_1, \boldsymbol{f}_2) = 1 - |\boldsymbol{f}_1 \cdot \boldsymbol{f}_2|. \tag{2}$$

Given this setup, we want features corresponding to pixels from the same object to be parallel, and features from different objects to be orthogonal. This is a powerful formulation with a continuous and piece-wise differentiable distance function between any two features. It allows for up to d objects with features maximally far apart. This is in contrast to setups that use directions [34] or bounded points [2] as features rather than lines, where there is only a single maximally distant feature to any other. The desired orthogonality of features from different segments plays a crucial role in defining our low-dimensional segment-space representation and in handling training data with incomplete labels, as we discuss later in this section. The output of our network is defined as a $h \times w \times d$ dimensional feature map, h and w being the height and width of the input image and we set d to be 128. We normalize the estimated features to unit length to get our per-pixel features $\boldsymbol{f} \in \mathcal{F}$. During training, we also define an L2 regularization loss that signals our network to generate unit-length features:

$$\mathcal{L}_u = \sum_i |1 - ||\boldsymbol{f}_i||_2|. \tag{3}$$

3.2   Determining Target Lines for each Entity

We aim to generate a feature for every pixel such that the feature of a pixel is parallel to others that belong in the same entity, and orthogonal to the ones that belong in others. This goal is shared with many standard metric or affinity learning formulations [3, 22, 34]. However, defining a loss function on all the inter-pixel affinities quickly becomes prohibitively expensive due to the quadratic explosion of the $N \times N$ possible pixel pairings, $N = h \times w$.

Instead, we first determine target lines for each entity during training and define losses that align each pixel's feature with its corresponding entity, while pushing it away from all other target lines. This simplifies our optimization problem from a many-to-many comparison setting to many-to-few. We will also use target lines to define our segmentation-space as described later in this section.

For each entity available in the ground-truth, we calculate the target line using the predicted features of all the pixels that belong to that entity. For homogeneous coordinates, euclidean averaging of the features may result in a degenerate average. Instead, we compute the average orientation $\boldsymbol{\mu}_k$ through

$$\boldsymbol{\mu}_k = \arg\max_{\boldsymbol{v}} \boldsymbol{v}^T M_k \boldsymbol{v}, \quad M_k = \sum_{i \in \mathcal{E}_k} \boldsymbol{f}_i \boldsymbol{f}_i^T, \tag{4}$$

where $\mathcal{E}_k$ is the set of pixels belonging to the ground-truth entity k. The solution of this maximization problem is given by the eigenvector of M_k corresponding to its largest eigenvalue. In line with our feature space definition in Eq. 1, $\boldsymbol{\mu}_k$ is of unit-length and changing its sign, or the sign of any $\boldsymbol{f}_i$ does not affect the result. Markley *et al.* [45] presents a comprehensive exposition of this orientation averaging approach in the case of quaternions.

3.3   Handling Imperfect Ground-truth

Most large-scale datasets with annotated ground-truth segmentation have been collected for class-based segmentation approaches such as semantic or panoptic segmentation. Due to the inherently limited set of classes a dataset contains, many objects that are not in one of the pre-defined classes are not segmented but either included in a general *background* category or just lack any label. Due to the complexity of annotating every single object in an image, even the class-agnostic SA-1B Dataset [33] contains many unlabeled objects.

As the goal of our class-agnostic segmentation approach is to generalize to any object, we can not treat the background category as its own entity. There may be multiple objects in the background category and without individual ground-truth labels, we can not determine a target line to align all the pixels. However, as an ideal representation in our feature space has orthogonal lines for each entity, we know that we want all the features in the background category to be orthogonal to the target lines $\boldsymbol{\mu}_k$ for all known entities in the image. Hence, when formulating our loss functions, we will exclude the background category for all losses promoting alignment and include them in ones that promote orthogonality. This way, we promote features in the background that are distinguishable from the known entities while not punishing the network for correctly estimating entities that are not annotated in the ground-truth.

Due to the complexity of annotating many segments with pixel-perfect precision, segmentation datasets often have boundary inaccuracies in the ground-truth labels. As a result, including the possibly inaccurate boundaries in the loss formulation harms the boundary accuracy of the system. To address this, we erode all the ground-truth annotations with a 5×5 kernel and exclude the eroded-away pixels from all loss computations.

3.4   Attraction and Repulsion

To align the features of the pixels that belong in a single entity to their target line $\boldsymbol{\mu}$, we define a simple *attraction* loss:

$$\mathcal{L}_a = \frac{1}{K} \sum_k \frac{1}{|\mathcal{E}_k|} \sum_{i \in \mathcal{E}_k} \left(1 - |\boldsymbol{f}_i \cdot \boldsymbol{\mu}_k|\right), \qquad (5)$$

where K is the number of labeled entities in the image. Rather than pairwise aligning all features, this simple loss pulls every pixel in an entity towards alignment with the same target line.
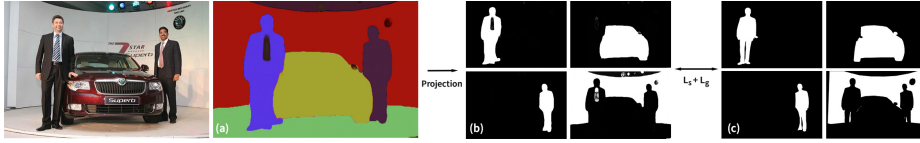
Fig. 2: We show our resulting feature map in (a), reduced with PCA and colorized for visualization. Our projection into the segment space results in a set of binary maps in (b) that are compared against the ground-truth in (c) via our segmentation loss.

Similarly, we define a *repulsion* loss that pushes every pixel to be orthogonal to the target lines of other entities:

$$\mathcal{L}_r = \frac{1}{\sqrt{K+1}} \left(\mathcal{L}_r^{BG} + \sum_k \frac{1}{|\mathcal{E}_k|} \sum_{l \neq k} \sum_{i \in \mathcal{E}_k} |\boldsymbol{f}_i \cdot \boldsymbol{\mu}_l| \right) \tag{6}$$

As noted in Sec. 3.3, using an average orientation $\boldsymbol{\mu}_{BG}$ of the background features, we include the background pixels in the repulsion loss:

$$\mathcal{L}_r^{BG} = \sum_k \frac{1}{|\mathcal{E}_k|} \sum_{i \in \mathcal{E}_k} |\boldsymbol{f}_i \cdot \boldsymbol{\mu}_{BG}| + \frac{1}{|BG|} \sum_{i \in BG} \sum_k |\boldsymbol{f}_i \cdot \boldsymbol{\mu}_k|,$$

that pushes the pixels in the background category to be orthogonal to all entity target lines, as well as the features in known entities to be orthogonal to the primary background orientation. We found that normalizing the repulsion loss with $K+1$ makes it ineffective for images with many segments. Instead, we use $\sqrt{K+1}$, providing a good balance for images with few or many entities.

During training, our network may fail to differentiate between two different entities and generate similar features for both. In such a case with $\boldsymbol{\mu}_k \approx \boldsymbol{\mu}_l$, the attraction and repulsion losses cancel each other. This results in a lack of a loss that signals the network to separate the two entities from each other. To promote separation between entities, we add the sparse regional contrast loss $\mathcal{L}_{rc}$ introduced by Liu *et al.* [41] as a second contrastive supervision, empirically setting the temperature $\tau = 0.5$ and using 256 queries per entity.

3.5 Segment-space Representation

In order to directly evaluate the clustering performance of the generated features, we compute per-entity segmentation maps by defining a smooth linear transformation from our feature-space to what we call the *segment-space*. Given K known entities excluding the background, we define segment-space as the $(K-1)$-dimensional projective sphere, as represented by the unit sphere in $\mathbb{R}^K$ with $\boldsymbol{\mu}_k$ serving as basis vectors. We define a dimension-reducing linear transformation matrix P from feature-space to segment-space such that segment k is represented by the kth unit vector $\boldsymbol{e}_k$:

$$P\boldsymbol{\mu}_k = \boldsymbol{e}_k \quad \forall k \tag{7}$$

Fig. 3: Two predictions from early training phases, (a) with only contrastive supervision and (b) with our segmentation loss added.

which we compute as the left-inverse of the matrix $A = [\boldsymbol{\mu}_1 \quad \boldsymbol{\mu}_2 \quad \cdots \quad \boldsymbol{\mu}_K]$ so long as such an inverse exists. In the case that the network fails to differentiate between two entities during training, i.e. $\boldsymbol{\mu}_k \approx \boldsymbol{\mu}_l$ for any k and l, the solution becomes degenerate. In such cases, we exclude our segmentation-space losses from back-propagation[4]. By transforming the feature vectors for each pixel to this sub-space with $P \cdot \boldsymbol{f}_i, \forall i$, we get a $h \times w$ map with k channels where the absolute value of the kth channel is a real-valued map in $[0, 1]$ that represents the alignment of each feature to $\boldsymbol{\mu}_k$. We will denote each channel in our segment-space with S_k. As the transformation P makes all target lines orthogonal, the losses we define in the segment-space are amplified within the space between any not-yet-orthogonal $\boldsymbol{\mu}_k$ and $\boldsymbol{\mu}_l$. This helps sort out features between entities that are not yet fully differentiated. As all background features should be orthogonal to all entities, their features should lie squarely in the null-space of P and their projections can be pushed towards $\mathbf{0}$.

As we aim to estimate features $\boldsymbol{f}_i$ that are aligned to $\boldsymbol{\mu}_k$ for $i \in \mathcal{E}_k$ and orthogonal for $i \notin \mathcal{E}_k$, the ground-truth for S_k, which we denote as $\hat{S}_k$, is the binary ground-truth segmentation map for the kth entity. We define our *segmentation loss* as the mean-squared error over each channel together with a gradient-based loss defined over multiple scales [36] to enforce spatial smoothness:

$$\mathcal{L}_s = \frac{1}{K} \sum_k MSE\left(S_k, \hat{S}_k\right), \qquad \mathcal{L}_g = \frac{1}{K} \sum_k \sum_m MSE\left(\nabla S_k^m, \nabla \hat{S}_k^m\right), \quad (8)$$

where ∇S_k^m is the spatial gradient of S_k at the mth scale. We visualize the projection and segment supervision in Figure 2. Transforming our features into our segment-space representation allows us to use this standard multi-scale gradient loss defined on single channels for promoting spatial consistency in our hyper-dimensional features that lie in projective space by back-propagating this loss through the linear mapping P. We show the effect of our segmentation loss in Figure 3 on the outputs of two toy networks after 8 epochs in training, one of which is trained exclusively on our contrastive losses, the other with the addition of the segmentation and multi-scale gradient losses. The contrastive losses enable the network to distinguish visual elements, while the additional segmentation losses enforce smooth features and turn the clusters into meaningful segments.

---

[4] These cases appear in less than 0.02% of iterations during early stages of training and their frequency drops with further training.

3.6   Network Architecture and Training

We define our final loss function as:

$$\mathcal{L} = \mathcal{L}_a + \mathcal{L}_r + \mathcal{L}_s + \lambda_{rc}\mathcal{L}_{rc} + \lambda_g\mathcal{L}_g + \lambda_u\mathcal{L}_u, \tag{9}$$

where λ_{rc}, λ_g, λ_u are set to be 0.125, 0.025, and 0.05, respectively. We employ the encoder-decoder architecture proposed by [62] as our feature generator, but replace the encoder with the base ConvNext [42] as the backbone. Following Ranftl *et al.* [53], we add four chained RefineNet [39] modules as decoder block, followed by three convolutional layers to upscale the features back to the input size. A final `tanh` activation generates the output features.

4   Inference-time Clustering

Our network is designed to generate pixel features that are aligned together for pixels that belong to a single entity. Our training process involves utilizing repulsion and segmentation losses to ensure that features associated with a specific entity are orthogonal to those of other entities. This per-pixel representation effectively distinguishes between different objects, facilitating the application of a simple clustering method for segmentation.

In our approach, the classical mean-shift clustering method aligns seamlessly with our mean-based representation. Specifically, we apply mean-shift clustering on the d-dimensional hyper-sphere with a bandwidth set to $\frac{\sqrt{2}}{2}$, representing a 45° separation. This choice leverages the orthogonal inter-entity features to cluster the image into distinct segments.

4.1   Multi-resolution Refinement

In our proof-of-concept implementation, we leverage a simple convolutional neural network (CNN) for its effectiveness in achieving robust training, even with small datasets, rendering it well-suited for our application. The inherent limitation of CNNs, where their reasoning capability is confined to the size of their receptive field, manifests in coherent outcomes at this resolution. However, as demonstrated in other mid-level vision tasks [7, 46], the capacity to generate intricate details significantly improves when inference is conducted at higher resolutions. This enhancement, however, comes at the cost of global coherency, as at higher resolutions, the network can only reason about small patches in the image at once, over-segmenting larger ones.

To capitalize on the consistency at smaller resolutions and the capacity to generate intricate segmentations at higher resolutions, we implement a multi-resolution clustering strategy. This involves feature generation and mean-shift clustering at multiple resolutions. The final segmentation map is constructed by processing segments from the smallest resolution first. With each increasing resolution, we incorporate clusters contained within existing segments in our map. New segments with a very high overlap with an old segment represent a refined

version of the same segment with higher boundary accuracy, while segments that subdivide existing segments reveal smaller objects detected in the image. Our low-to-high resolution cluster merging strategy enables the generation of highly detailed segmentation maps in in-the-wild images with varying contexts. We provide a detailed description of our multi-resolution refinement approach in the supplementary material.

5   Experiments

We train our model for 60 epochs with a learning rate of 1×10^{-5}. We use ImageNet [17]-pretrained weights for the encoder and train all other modules from scratch. We use the class-based COCO Panoptic [5], ADE20K [66], and CIHP [21] datasets as well as procedurally generates images of simple geometric shapes for training. We give a detailed description of our training process and extend our experimental analysis in the supplementary material.

Class-agnostic Baselines We evaluate our method against recent class-agnostic segmentation methods Open-World Entity Segmentation (OWES) [52], High-Quality Entity Segmentation (HQES) [51], and the Segment Anything (SAM) model [33] with the smaller ViT-B encoder as it has a similar number of parameters to ours. Our method and OWES [52] use common class-based datasets as real-world training data. HQES [51] trains solely on their novel high-resolution class-agnostic dataset of 600K annotations in 33K images targeting high recall in-the-wild. SAM [33] trains their model on their novel large-scale dataset of 1B annotations in 11M images in addition to the common class-based datasets.

Class-based Baselines We also include several panoptic segmentation methods in our analysis, namely OneFormer [27] and Mask2Former [13] as task-unifying segmentation approaches, MaskDINO [35] being optimized for object detection, and kMaX-DeepLab [65] inspired by k-means clustering. As class-based approaches do not allow for training with multiple datasets due to class definition conflicts, we use their models trained on ADE20k [66].

Metrics We compute Recall as a measure of completeness, treating every segment as a true positive that has an IoU > 0.5 with a ground truth segment. Recall being a critical measure of performance for class-agnostic evaluation, we also report the percentage relative improvement $\Delta\%$ of all methods with respect to the baseline with the lowest recall. As a measure of segmentation accuracy, we use the standard mask-based intersection-over-union (IoU) metric as well as the Boundary IoU metric [12] that focuses on boundary accuracy of the segments. The class-agnostic segmentation methods including ours, OWES [52], HQES [51], and SAM [33], will naturally generate segments that may not be included in the ground-truth maps, as it is very challenging to annotate every object in every scene. This makes the evaluation over false positives not a meaningful metric for this task, hence we exclude it.

5.1   Evaluation over a Class-based Train/Test Split

Class-agnostic segmentation aims to generalize to objects of any class, characteristically being trained on multiple class-based datasets or large class-agnostic datasets. Class-based segmentation, on the other hand, focuses on a specific set of classes in a fixed domain of images. In order to measure the effect of class-agnostic generalization on the performance on known classes, we present an evaluation in the domain of class-based segmentation methods using the test split of ADE20K [66] in Table 1. All methods in Table 1 use the training split of ADE20K in their training.

Our method with the multi-resolution refinement performs on-par with the class-based OneFormer [27] despite our smaller architecture, with a slight improvement in recall.

Table 1: Evaluation on ADE20k [66]

Method	Architecture-#Params	mIoU↑	B. IoU ↑	Recall↑	Δ%↑
kMaX-Deeplab [65]	ConvNext-L - 244m	0.366	0.332	0.376	0
MaskDINO [35]	Swin-L - 223m	0.373	0.340	0.387	3
Mask2Former [13]	Swin-L - 216m	0.413	0.375	0.430	14
OneFormer [27]	Swin-L - 219m	0.455	**0.418**	0.484	29
OWES [52]	Swin-L - 208m	0.400	0.357	0.420	12
Ours - base res.	ConvNext-B - 101m	0.380	0.327	0.391	4
Ours	ConvNext-B - 101m	**0.462**	0.407	**0.494**	**31**

This shows that our clustering-based bottom-up approach does not lead to a drop in performance on known classes. We see a drop in performance for the class-agnostic baseline OWES despite it having seen these classes during training, pointing to the mixed dataset training in top-down approaches having an adverse effect in domain-specific scenarios.

5.2   Zero-shot Class-agnostic Evaluation

Reflecting the in-the-wild generalization motivation of class-agnostic segmentation, we perform zero-shot evaluations on two recent class-agnostic datasets EntitySeg [51] and SA-1B [33] that are characterized by their high number of class-agnostic annotations per image and high-resolution input images. As these datasets were used to train the models of HQES [51] and SAM [33], respectively, we exclude these methods from corresponding tables.

We present our evaluation on the first subset of the SA-1B dataset [33] in Table 2. We use the first 1000 images in the dataset for general performance, and also create a different split of 500 images with the highest number of annotated objects in the set to measure the true positive rates in complex scenes. As Table 2 demonstrates, we significantly improve over class-agnostic approaches OWES and HQES,

Table 2: Zero-shot evaluation on SA-1B [33]

Method	General performance				High object count			
	mIoU↑	B.IoU ↑	Recall↑	Δ%↑	mIoU↑	B.IoU ↑	Recall↑	Δ%↑
Mask2Former [13]	0.318	0.306	0.326	0	0.263	0.285	0.274	0
OneFormer [27]	0.335	0.339	0.342	5	0.294	0.357	0.305	11
kMaX-Deeplab [65]	0.337	0.337	0.344	6	0.295	0.338	0.308	12
MaskDINO [35]	0.341	0.340	0.348	7	0.316	0.364	0.329	20
OWES [52]	0.356	0.338	0.370	14	0.349	0.380	0.366	34
HQES [51] (CF/Swin-L/217 m)	0.398	0.418	0.411	26	0.384	0.448	0.401	46
Ours	**0.457**	**0.424**	**0.500**	**53**	**0.480**	**0.460**	**0.529**	**93**

doubling the improvement in recall of second-best HQES with respect to the Mask2Former [13] baseline. Our performance comes despite our smaller architecture as well as the high-resolution class-agnostic dataset collected to train HQES

for in-the-wild generalization. This shows the effectiveness of our bottom-up approach in generating a class-agnostic understanding of objectness from class-based datasets. We observe an expected drop in performance for class-based approaches, as class-agnostic datasets naturally contain many objects that are outside the list of classes for which these networks are trained.

Table 3 presents our evaluation on the EntitySeg dataset [51]. We perform on-par with SAM [33] despite their use of a class-agnostic dataset that is larger than our training set by two magnitudes. This demonstrates that our carefully designed clustering-based approach is highly effective in generalizing to in-the-wild images without the need for a large-scale dataset.

Earlier attempts at class-agnostic segmentation [25, 52, 63] focused on developing classless formulations and trained on

Table 3: Zero-shot on EntitySeg [51]

Method	mIoU↑	B. IoU ↑	Recall↑	Δ%↑
OneFormer [27]	0.409	0.382	0.423	0
Mask2Former [13]	0.452	0.403	0.474	12
MaskDINO [35]	0.463	0.423	0.482	14
kMaX-Deeplab [65]	0.488	0.455	0.524	24
OWES [52]	0.521	0.470	0.566	34
SAM [33]	**0.585**	**0.539**	**0.619**	**46**
Ours	<u>0.574</u>	<u>0.522</u>	<u>0.614</u>	<u>45</u>

available class-based datasets. This focus shifted to the collection of high-quality and large-scale datasets in recent literature [33,51] due to the straightforward effectiveness of a well-crafted training dataset in generalization despite their cost. Our methodology stands orthogonal to the recent literature, where we achieve state-of-the-art performance in in-the-wild class-agnostic segmentation using a novel bottom-up approach to the segmentation problem using limited training data and a simple network architecture. While this divergence creates new promising research and development directions for class-agnostic segmentation, combining our clustering-based formulation with large-scale training procedures, it also enables the application of our methodology in other, data-scarce segmentation problems as discussed in the next section.

5.3   Evaluation on Cell and Nucleus Segmentation

Our bottom-up approach is designed with a focus on simply generating orthogonal features for differentiating segments in an image. Our formulation being developed towards a simple clustering of features achieves state-of-the-art performance even when trained on a smaller

Table 4: Evaluation on EVICAN dataset [55]

Method	Easy Difficulty			Medium Difficulty			Hard Difficulty		
	mAP↑	AP@50↑	AP@75↑	mAP↑	AP@50↑	AP@75↑	mAP↑	AP@50↑	AP@75↑
MRCNN [55]	0.322	0.616	0.317	0.136	0.310	0.105	0.085	0.208	0.044
DeepCeNS [29]	**0.526**	**0.834**	**0.573**	<u>0.261</u>	<u>0.479</u>	**0.289**	<u>0.169</u>	<u>0.338</u>	<u>0.158</u>
Ours	<u>0.408</u>	<u>0.663</u>	<u>0.400</u>	**0.304**	**0.565**	<u>0.223</u>	**0.322**	**0.589**	**0.290**

training dataset. This makes our method directly applicable to segmentation problems in other domains such as biomedical images. We demonstrate this by training our network, without any changes in the formulation, on the EVICAN dataset [55] for the problem of cell and nucleus segmentation. EVICAN dataset provides 4464 annotated microscopic images in their training split, where images are typically of very little contrast. Using their test set, divided by them into 3 difficulty levels, we compare against domain-specific biomedical segmentation

approaches MRCNN [55] and DeepCeNS [29] in Table 4. Following Schwendy *et al*. [55], we use the average precision (AP) with different IoU thresholds. We report the AP at 50 and 75% IoU, as well as the mean value of AP's at 10 different IoU thresolds between 50-95%. We show competitive performance in easy and medium difficulties, while significantly improving upon the baselines in the hard difficulty subset. This demonstrates the generic nature of our bottom-up approach to segmentation.

The network architectures adopted by these domain-specific approaches are both convolutional neural networks similar to ours. Although CNN's have inherent limitations coming from a fixed receptive field, their ease of training makes them applicable to data-scarce domains. This is in contrast with transformer-based architectures such as HQES [51] and SAM [33] that can not be trained using such small datasets. While our formulation can be applied to transformer-based architectures, we show that the receptive field limitation of CNN's can be remedied within our feature-based formulation with a simple multi-resolution estimation procedure as detailed in Section 4.1.

5.4   The Effect of the Segment-space Representation

Our system makes use of metric-learning inspired attraction and repulsion losses $\mathcal{L}_a$ and $\mathcal{L}_c$ that allows the network to differentiate between entities. We also include the regional contrast loss $\mathcal{L}_{rc}$ for cases where the network groups two entities together, as detailed in Section 3.4. In order to improve clustering quality, we develop our segment-space representation and define an MSE and a gradient-based loss, $\mathcal{L}_s$ and $\mathcal{L}_g$, respectively. In this section, we measure the effect of our segment-space representation on our generated features. For this purpose, we measure the two qualities we expect from the generated features: inter-mean similarity and intra-entity similarity. Inter-mean similarity measures if the network is able to generate different mean orientations $\boldsymbol{\mu}_k$ for different entities, computed as the average cosine similarity between each $(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l)$ pair in an image.

If the network is able to generate perfectly orthogonal orientations for every entity in the image, our inter-mean similarity metric would be 0. Intra-entity similarity is computed as the average cosine similarity between

Table 5: We compute the cosine-similarity between means of different entities as well as features within and entity for different loss combinations on the ADE20K [66] dataset.

Included losses	Inter-mean sim.↓ (deg. ↑)		Intra-entity sim.↑ (deg. ↓)	
$\mathcal{L}_{rc} + \mathcal{L}_a + \mathcal{L}_r$	**0.024**	(88.6°)	0.152	(81.2°)
$\mathcal{L}_{rc} + \mathcal{L}_a + \mathcal{L}_r + \mathcal{L}_s$	0.031	(88.2°)	<u>0.745</u>	(41.8°)
$\mathcal{L}_{rc} + \mathcal{L}_a + \mathcal{L}_r + \mathcal{L}_s + \mathcal{L}_g$	<u>0.029</u>	(88.3°)	**0.754**	(41.1°)

the mean assigned to each entity and the features generated for pixels belonging to that entity. It measures how well-aligned the features belonging to an entity to their corresponding mean orientation $\boldsymbol{\mu}_k$. In the case perfect alignment, our intra-entity similarity metric would be 1. A good performance in both metrics is desired for effective clustering of our features.

We compare the improvement brought by $\mathcal{L}_s$ and $\mathcal{L}_g$ in Table 5. We perform this test by training 3 different versions of our formulation on the ADE20k

training set for 10 epochs. It should be noted that all 3 versions include our differentiation-based losses. This is required for our network to generate different features for different entities, as our segment-space based losses only promote better alignment inside a given entity. As Table 5 shows, our discriminative losses are effective in generating almost perfectly orthogonal orientations for different entities in all scenarios, while struggling to generate well-aligned features for each segment. Our segment-space loss $\mathcal{L}_s$ significantly improves the performance in this aspect, driving intra-entity similarity below the critical 45°threshold that represents clear separation between the features of two entities with orthogonal orientations. Our gradient-based loss $\mathcal{L}_g$ further improves the performance both in terms of inter-mean orthogonality and intra-entity alignment.

6   Conclusion and Limitations

In this paper, we introduced a bottom-up approach to class-agnostic segmentation and show that this generalist approach shows a great generalization ability to out-of-distribution images despite using standard class-based segmentation datasets as real-world training data. This is in line with our current understanding of human cognition which is modeled as a bottom-up process. Our network is trained with a novel formulation that integrates ideas from contrastive learning with our new segment-space representation to achieve detailed clustering of semantically meaningful regions in any image. We demonstrate the performance of our formulation using a proof-of-concept implementation with a small convolutional architecture trained on around 200k real-world images with class-based annotations. Our approach represents a promising new direction for in-the-wild class-agnostic segmentation that is typically approached with top-down formulations as well as diverse segmentation challenges in other domains with limited training data. We make use of iterative clustering of objects on image patches to generate high-resolution segmentation results. This circumvents the limited receptive field size of convolutional neural networks. The performance of our method can be further improved by utilizing network architectures with higher number of parameters and higher receptive field size.

We so far have only cared for entities to be different, using the segments included in class-based datasets as individual entities. However, this entity definition does not reflect the complexity of the real world. The definition of an entity varies depending on the context. For instance, while the common datasets treat *person* as a single entity, there are many applications that would benefit from the segmentation of body parts or facial features. We believe a hierarchical representation of entities will allow class-agnostic segmentation to be applicable to a wider range of problems.

Acknowledgements

References

1. Atigh, M.G., Schoep, J., Acar, E., Van Noord, N., Mettes, P.: Hyperbolic image segmentation. In: Proc. CVPR (2022)
2. Bai, M., Urtasun, R.: Deep watershed transform for instance segmentation. In: Proc. CVPR (2017)
3. Banerjee, A., Dhillon, I.S., Ghosh, J., Sra, S., Ridgeway, G.: Clustering on the unit hypersphere using von mises-fisher distributions. Journal of Machine Learning Research **6**(9) (2005)
4. Biederman, I.: Recognition-by-components: A theory of human image understanding. Psychological Review **94**(2), 115–147 (1987)
5. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: Proc. CVPR (2018)
6. Cai, Z., Vasconcelos, N.: Cascade r-cnn: High quality object detection and instance segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **43**(5), 1483–1498 (2021)
7. Careaga, C., Aksoy, Y.: Intrinsic image decomposition via ordinal shading. ACM Trans. Graph. (2023)
8. Cetin, E., Chamberlain, B.P., Bronstein, M.M., Hunt, J.J.: Hyperbolic deep reinforcement learning. In: Proc. ICLR (2023)
9. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal. Mach. Intell. **40**(4), 834–848 (2017)
10. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: Proc. ICML (2020)
11. Cheng, B., Collins, M.D., Zhu, Y., Liu, T., Huang, T.S., Adam, H., Chen, L.C.: Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In: Proc. CVPR (2020)
12. Cheng, B., Girshick, R., Dollár, P., Berg, A.C., Kirillov, A.: Boundary IoU: Improving object-centric image segmentation evaluation. In: CVPR (2021)
13. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: Proc. CVPR (2022)
14. Cheng, B., Schwing, A.G., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation. In: Proc. NeurIPS (2021)
15. Cheng, H.D., Sun, Y.: A hierarchical approach to color image segmentation using homogeneity. IEEE Trans. Pattern Anal. Mach. Intell. **9**(12), 2071–2082 (2000)
16. De Brabandere, B., Neven, D., Van Gool, L.: Semantic instance segmentation for autonomous driving. In: Proc. CVPR Workshops (2017)
17. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Proc. CVPR (2009)
18. Fathi, A., Wojna, Z., Rathod, V., Wang, P., Song, H.O., Guadarrama, S., Murphy, K.P.: Semantic Instance Segmentation via Deep Metric Learning. arXiv preprint arXiv:1703.10277 [cs] (2017)
19. Gao, N., Shan, Y., Wang, Y., Zhao, X., Yu, Y., Yang, M., Huang, K.: Ssap: Single-shot instance segmentation with affinity pyramid. In: Proc. ICCV (2019)
20. Ge, S., Mishra, S., Kornblith, S., Li, C.L., Jacobs, D.: Hyperbolic contrastive learning for visual representations beyond objects. In: Proc. CVPR (2023)
21. Gong, K., Liang, X., Li, Y., Chen, Y., Yang, M., Lin, L.: Instance-level human parsing via part grouping network. In: Proc. ECCV (2018)
22. Gopal, S., Yang, Y.: Von mises-fisher clustering models. In: International Conference on Machine Learning. pp. 154–162. PMLR (2014)

23. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proc. CVPR (2020)
24. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask r-cnn. IEEE Trans. Pattern Anal. Mach. Intell. **42**, 386–397 (2020)
25. Hwang, J., Oh, S.W., Lee, J.Y., Han, B.: Exemplar-based open-set panoptic segmentation network. In: Proc. CVPR (2021)
26. Hwang, J.J., Yu, S.X., Shi, J., Collins, M.D., Yang, T.J., Zhang, X., Chen, L.C.: Segsort: Segmentation by discriminative sorting of segments. In: Proc. ICCV (October 2019)
27. Jain, J., Li, J., Chiu, M., Hassani, A., Orlov, N., Shi, H.: OneFormer: One Transformer to Rule Universal Image Segmentation. In: Proc. CVPR (2023)
28. Ji, S., Park, H.W.: Image segmentation of color image based on region coherency. In: Proc. ICIP (1998)
29. Khalid, N., Munir, M., Edlund, C., Jackson, T.R., Trygg, J., Sjögren, R., Dengel, A., Ahmed, S.: Deepcens: An end-to-end pipeline for cell and nucleus segmentation in microscopic images. In: Proc. IJCNN. IEEE (2021)
30. Khrulkov, V., Mirvakhabova, L., Ustinova, E., Oseledets, I., Lempitsky, V.: Hyperbolic image embeddings. In: Proc. CVPR (2020)
31. Kirillov, A., Girshick, R., He, K., Dollár, P.: Panoptic feature pyramid networks. In: Proc. CVPR (2019)
32. Kirillov, A., He, K., Girshick, R., Rother, C., Dollár, P.: Panoptic segmentation. In: Proc. CVPR (2019)
33. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollar, P., Girshick, R.: Segment anything. In: Proc. ICCV (2023)
34. Kong, S., Fowlkes, C.C.: Recurrent pixel embedding for instance grouping. In: Proc. CVPR (2018)
35. Li, F., Zhang, H., xu, H., Liu, S., Zhang, L., Ni, L.M., Shum, H.Y.: Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In: Proc. CVPR (2023)
36. Li, Z., Snavely, N.: Megadepth: Learning single-view depth prediction from internet photos. In: Proc. CVPR (2018)
37. Li, Z., Wang, W., Xie, E., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P., Lu, T.: Panoptic segformer: Delving deeper into panoptic segmentation with transformers. In: Proc. CVPR (2022)
38. Lin, F., Bai, B., Guo, Y., Chen, H., Ren, Y., Xu, Z.: Mhcn: A hyperbolic neural network model for multi-view hierarchical clustering. In: Proc. ICCV (2023)
39. Lin, G., Liu, F., Milan, A., Shen, C., Reid, I.: Refinenet: Multi-path refinement networks for dense prediction. IEEE Trans. Pattern Anal. Mach. Intell. (2019)
40. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Proc. ECCV (2014)
41. Liu, S., Zhi, S., Johns, E., Davison, A.: Bootstrapping semantic segmentation with regional contrast. In: Proc. ICLR (2022)
42. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: Proc. CVPR (2022)
43. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proc. CVPR (2015)
44. Luo, J., Gray, R.T., Lee, H.C.: Incorporation of derivative priors in adaptive bayesian color image segmentation. In: Proc. ICIP (1998)
45. Markley, F.L., Cheng, Y., Crassidis, J.L., Oshman, Y.: Averaging quaternions. Journal of Guidance, Control, and Dynamics **30**(4), 1193–1197 (2007)

46. Miangoleh, S.M.H., Dille, S., Mai, L., Paris, S., Aksoy, Y.: Boosting monocular depth estimation models to high-resolution via content-adaptive multi-resolution merging. In: Proc. CVPR (2021)
47. Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., Terzopoulos, D.: Image segmentation using deep learning: A survey. IEEE Trans. Pattern Anal. Mach. Intell. **44**(7), 3523–3542 (2021)
48. Nickel, M., Kiela, D.: Poincaré embeddings for learning hierarchical representations. NeurIPS **30** (2017)
49. Nock, R., Nielsen, F.: Statistical region merging. IEEE Trans. Pattern Anal. Mach. Intell. (2004)
50. Qi, L., Kuen, J., Lin, Z., Gu, J., Rao, F., Li, D., Guo, W., Wen, Z., Yang, M.H., Jia, J.: Ca-ssl: Class-agnostic semi-supervised learning for detection and segmentation. In: Proc. ECCV (2022)
51. Qi, L., Kuen, J., Shen, T., Gu, J., Guo, W., Jia, J., Lin, Z., Yang, M.H.: High quality entity segmentation. In: Proc. ICCV (2023)
52. Qi, L., Kuen, J., Wang, Y., Gu, J., Zhao, H., Torr, P., Lin, Z., Jia, J.: Open world entity segmentation. IEEE Trans. Pattern Anal. Mach. Intell. (2022)
53. Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., Koltun, V.: Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. IEEE Trans. Pattern Anal. Mach. Intell. **44**(3), 1623–1637 (2020)
54. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Proc. MICCAI (2015)
55. Schwendy, M., Unger, R.E., Parekh, S.H.: Evican—a balanced dataset for algorithm development in cell and nucleus segmentation. Bioinformatics **36**(12), 3863–3870 (2020)
56. Tian, Z., Shen, C., Chen, H.: Conditional convolutions for instance segmentation. In: Proc. ECCV (2020)
57. Tremeau, A., Borel, N.: A region growing and merging algorithm to color segmentation. Pattern recognition **30**(7), 1191–1203 (1997)
58. Uhrig, J., Rehder, E., Fröhlich, B., Franke, U., Brox, T.: Box2pix: Single-shot instance segmentation by assigning pixels to object boxes. In: 2018 IEEE Intelligent Vehicles Symposium (IV). IEEE (2018)
59. Wang, X., Zhang, R., Kong, T., Li, L., Shen, C.: Solov2: Dynamic and fast instance segmentation. In: Proc. NeurIPS (2020)
60. Wang, X., Zhang, R., Shen, C., Kong, T., Li, L.: Solo: A simple framework for instance segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **44**(11), 8587–8601 (2021)
61. Weng, Z., Ogut, M.G., Limonchik, S., Yeung, S.: Unsupervised discovery of the long-tail in instance segmentation using hierarchical self-supervision. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2603–2612 (2021)
62. Xian, K., Shen, C., Cao, Z., Lu, H., Xiao, Y., Li, R., Luo, Z.: Monocular relative depth perception with web stereo data supervision. In: Proc. CVPR (2018)
63. Xu, H.M., Chen, H., Liu, L., Yin, Y.: Dual decision improves open-set panoptic segmentation. In: Proc. BMVC (2022)
64. Yi-de, M., Qing, L., Zhi-Bai, Q.: Automated image segmentation using improved pcnn model based on cross-entropy. In: Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, 2004. IEEE (2004)
65. Yu, Q., Wang, H., Qiao, S., Collins, M., Zhu, Y., Adam, H., Yuille, A., Chen, L.C.: k-means mask transformer. In: Proc. ECCV (2022)

66. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proc. CVPR (2017)
67. Zhou, T., Wang, W., Konukoglu, E., Van Gool, L.: Rethinking semantic segmentation: A prototype view. In: Proc. CVPR (2022)