EXACT: COMPOSITIONAL AUGMENTATION FOR IMAGE-LEVEL WEAKLY-SUPERVISED INSTANCE SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose EXACT: EXtract-AugContext-pasTe, a compositional image augmentation pipeline for weakly-supervised instance segmentation using only image-level supervision. The proposed method consists of three main components. The first component generates high-quality foreground object masks. To this end, an EM-like approach is proposed that iteratively refines an initial set of object mask proposals generated by a generic entity segmentation method. Next, in the second component, high-quality context-aware background images are generated using a text-to-image compositional synthesis method like DALL·E. Finally, the third component creates a large-scale pseudo-labeled instance segmentation training dataset by compositing the foreground object masks onto the original and generated background images. The proposed approach achieves state-of-the-art weakly-supervised instance segmentation results on both the PASCAL VOC 2012 and MS COCO dataset by using only image-level, weak label information. In particular, it outperforms the best baseline by +7.4 and +2.8 $\text{mAP}_{0.50}$ on PASCAL and COCO, respectively. Further, the method provides a new solution to the long-tail weakly-supervised instance segmentation problem (when many classes may only have few training samples), by selectively augmenting under-represented classes.

1 INTRODUCTION

The instance segmentation task aims to assign an instance label to every pixel in an image. It has found many applications in many real-world domains (Hafiz & Bhat, 2020), e.g., self-driving cars, AR/VR, robotics, etc. Standard approaches to solving this problem involve framing it as a per-pixel labeling problem in a deep learning framework (He et al., 2017; Hafiz & Bhat, 2020).

Training of instance segmentation methods requires a vast amount of labeled data (He et al., 2017). Getting a large labeled dataset with per-pixel instance labels is very expensive, requires significant human effort, and is also a time-consuming process. In order to tackle these issues, alternative approaches have been proposed. One direction involves utilizing synthetic data to train instance segmentation methods (Richter et al., 2017; Hu et al., 2019). However, they generally suffer from the sim2real domain gap, and expert knowledge is required to create synthetic environments (Hodaň et al., 2019). A few other works have used object cut-and-paste concepts (Dwibedi et al., 2017; Ghiasi et al., 2021) to augment training data for instance segmentation tasks. However, these methods require availability of accurate foreground object masks, so that objects can accurately be cut before they are pasted. Acquiring these foreground masks may require extensive human efforts, which can make this line of work difficult to scale.

Weakly-supervised learning approaches have evolved as important alternatives to solving the problem. A few of these methods (Khoreva et al., 2017; Liao et al., 2019; Sun et al., 2020; Arun et al., 2020; Hsu et al., 2019) involve leveraging bounding boxes as a source of weak supervision. Bounding boxes contain important cues about object sizes and also their instance labels. However, even bounding box labels are taxing to label. Another line of works (Zhou et al., 2018; Zhu et al., 2019; Cholakkal et al., 2019; Ge et al., 2019; Hwang et al., 2021; Laradji et al., 2019; Kim et al., 2021; Ahn et al., 2019; Arun et al., 2020; Liu et al., 2020) explores using only image-level labels for learning instance segmentation. Due to lack of segmentation annotation, those works generally need to introduce object priors from region proposals (Zhou et al., 2018; Zhu et al., 2019; Arun et al., 2020; Laradji et al., 2019). One approach involves utilizing signals from class activation maps (Zhou et al., 2018; Zhu et al., 2019),

yet those maps do not provide strong instance-level information but only semantic-level, and can be noisy and/or not very accurate as well. Another procedure involves generating pseudo-label from proposals and training a supervised model with pseudo-label as ground truth. Yet those methods can not generate high quality pseudo-labels which hinder the supervised model performance.

In this work, we propose EXtract-AugContext-pasTe (EXACT), a new weakly-supervised instance segmentation approach using only image-level labels. Our contributions are: First, "EXtract": we extract high quality foreground object masks using an Expectation Maximization (EM)-like method to iteratively optimize the foreground mask distribution of each interested class and refine object mask proposals from generic region segmentation methods (Maninis et al., 2016), (Arbeláez et al., 2014), (Qi et al., 2021). Then "AugContext", we generate high-quality background images, by first captioning the source image, then passing them to text-to-image synthesis method, e.g., DALL-E (Ramesh et al., 2021; sbe; Ding et al., 2021). Finally, "pasTe": we create a large labeled training dataset by pasting the foreground masks onto the original and generated context images (Fig. 3).

We achieve state-of-the-art (SOTA) performance on weakly-supervised instance segmentation on the PASCAL VOC (Everingham et al., 2010) and COCO dataset (Lin et al., 2014a) using only image-level weak label. We outperform the best baselines by +7.3 and +2.8 $\text{mAP}_{0.50}$ on Pascal VOC and COCO datasets respectively. EXACT also provides a new solution to long-tail weakly-supervised instance segmentation problem on Pascal VOC dataset. Additionally, we also show that EXACT is generalizable to object detection task.

2 RELATED WORKS

Weakly Supervised Instance Segmentation Since acquiring per-pixel segmentation annotations is time-consuming and expensive, many weakly supervised methods have been proposed to utilize cheaper labels. Existing weakly supervised instance segmentation methods can be largely grouped in two categories, characterized by labels that the algorithms can access during the training phase. The first line of works explores the use of bounding boxes as weak labels for instance segmentation tasks (Khoreva et al., 2017; Liao et al., 2019; Sun et al., 2020; Arun et al., 2020; Hsu et al., 2019). Notably, Khoreva et al. (2017) generate pseudo-instance mask by `GrabCut+` and MCG (Arbeláez et al., 2014), and Hsu et al. (2019) restraint the bounding box by tightness. Another series of works have also started using image-level labels as weak labels for instance segmentation tasks (Zhou et al., 2018; Zhu et al., 2019; Cholakkal et al., 2019; Ge et al., 2019; Hwang et al., 2021; Laradji et al., 2019; Kim et al., 2021; Ahn et al., 2019; Arun et al., 2020; Liu et al., 2020). Notably, Zhou et al. (2018) utilizes class peak response, Ge et al. (2019) refines segmentation seed by a multi-task approach, Arun et al. (2020) improve the generated pseudo-labels by viewing them as conditional probabilities, and Kim et al. (2021) transfer semantic knowledge from semantic segmentation to obtain pseudo instance label. However, aggregating pseudo-label across multiple images remains largely unexplored.

Data Augmentations for Instance Segmentation In recent years, data augmentation has been an indispensable component in solving instance segmentation tasks (Hafiz & Bhat, 2020; He et al., 2017). Ghiasi et al. (2021) found that large-scale jittering plays an important role in learning a strong instance segmentation model, especially in a weakly-supervised setting. Dwibedi et al. (2017) proposed a new paradigm of data augmentation which augments the instances by rotation, scaling, and then pastes the augmented instances to images. Entitled cut paste augmentation strategy can diversify training data to a very large scale. Empirical experiments (Dwibedi et al., 2017; Ghiasi et al., 2021) have found that cut paste augmentation can lead to a major boost in instance segmentation datasets. These approaches require the presence of foreground object masks. So they can not be applied for weakly-supervised instance segmentation problems using only image-level labels. In contrast, our approach is designed to work with only image-level label information.

Long-Tail Visual Recognition Generally instance segmentation models fail to perform well in real-world scenarios due to the long-tail nature of object categories in natural images (Gupta et al., 2019; Van Horn et al., 2018; He & Garcia, 2009; Liu et al., 2019). A long-tail dataset consists of mostly head classes objects, while tail classes comprise relatively few instances. Existing instance segmentation methods (He et al., 2017) often yield poor performance on tail classes, and sometimes predict head class all the time (Wang et al., 2021). Existing methods to alleviate this include supervising models using a new loss that favors tail classes (Wang et al., 2021; Hsieh et al., 2021) and dataset balancing techniques (He & Garcia, 2009; Mahajan et al., 2018) that re-distribute classes so that model is able to see more tail class instances. However, few works evaluate weakly-supervised methods in the long-tail setting.

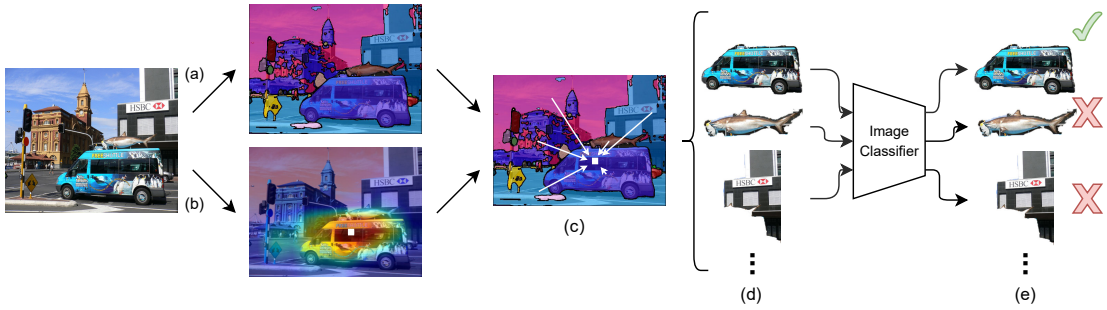


Figure 1: Step 1 of foreground extraction. (a) Entity Segmentation extracts segments from images. (b) Grad-CAM highlights a region based on the given label, and the center of moments (white dot on the image) is calculated for the highlighted region. (c) For all eligible segments, we compute the pixel-wise average distance to the center of the region highlighted by Grad-CAM. (d) We select n segments that have the shortest distances to the center. (e) All n foreground candidate segments are filtered using the classifier network, and we select the foreground with highest predicted probability.

3 METHOD

Our goal is to learn an instance segmentation model in a weakly supervised framework using only image-level labels. To this end, we propose EXACT (EXtract-AugContext-pasTe) approach that consists of three main components: foreground extraction (§3.1), background augmentation (§3.2), and compositional paste (§3.3). EXACT produces an augmented dataset with pseudo-labels, and we train a supervised model using pseudo-labels as ground-truth.

3.1 FOREGROUND EXTRACTION

We propose an expectation-maximization (EM) based foreground extraction (F-EM) algorithm. Given only image-level labels for a dataset, F-EM extracts as many high-quality foreground object masks as possible by iteratively optimizing the foreground mask distribution of each interested class and refining object mask proposals. There are three steps: 1) region proposal, 2) Maximization step to estimate object foreground distribution statistics of each interested object class, 3) Expectation step to refine the collection of matching region proposals given the approximated object foreground distribution statistics. Steps 2 and 3 are performed in an interactive manner. Fig. 2 demonstrates different steps for extraction of foreground object masks.

Step 1: Region Proposal. In this step, the goal is to generate candidate foreground object segments corresponding to a given image label for each image. Suppose we are given a dataset $\mathcal{D} = \{(I_i, \mathbf{y}_i)\}_{i=1}^{N}$ where each image I_i may contain one or more objects of different classes, therefore $\mathbf{y}_i$ is a binary vector that corresponds to image-level object labels for a multilabeled image I_i. We train an image classifier $f(\cdot)$ which takes image I as input and predicts image label: $\mathbf{y} = f(I)$. Given a ground truth class a, for each image I_i where $y_i^a = 1$, meaning that an object of class a is present in image I_i, we generate the Grad-CAM (Selvaraju et al., 2017) activation map through the classifier $f(\cdot)$ (Fig. 1 (b)). Then we threshold the activation map to convert it to a binary mask G_i^a that is associated with class a and calculate x-y coordinate of the center of gravity of the object mask as: $c_i^a = (c_{ix}^a, c_{iy}^a) = (\sum_{x,y} G_i^a(x,y)x / \sum_{x,y} G_i^a(x,y),\ \sum_{x,y} G_i^a(x,y)y / \sum_{x,y} G_i^a(x,y))$. c_i^a will be used as anchor to select foreground segments of class a object for image I_i.

Next, for each input image I_i, we use an off-the-shelf generic region proposal method to propose candidate objects. The generic region proposal methods include super-pixel methods (SLIC (Achanta et al., 2012), GCa10 (Veksler et al., 2010; Felzenszwalb & Huttenlocher, 2004)) and hierarchical entity segment methods (MCG (Arbeláez et al., 2014), COB(Maninis et al., 2016), entity segmentation (Qi et al., 2021)). These approaches only propose general class-agnostic segmentation masks with no class labels for the segments. In this work we show the results of using the entity segmentation method (Qi et al., 2021) and COB(Maninis et al., 2016) to obtain a set of segments of the image $\mathcal{S}_i = \{s_i^1, s_i^2, ..., s_i^m\}$, but we note that our method is compatible to other methods as well.

Then we use the above computed Grad-CAM location anchor c_i^a for interest class a in image I_i to find the correct foreground segment O_i^a with label a from $\mathcal{S}_i$. For each segment s_i^j, we calculate a pixel-wise average distance to the anchor c_i^a. We have the **location assumption** that the correct foreground segments should have a large overlap with the Grad-CAM mask G_i^a. In other words, foreground object mask O_i^a for object class a should have short average euclidean distance to the Grad-CAM center (dist(O_i^a, c_i^a)). We observe that this is generally true when there is only one object present in an image. But, in many images, more than one object from the same class can be present.

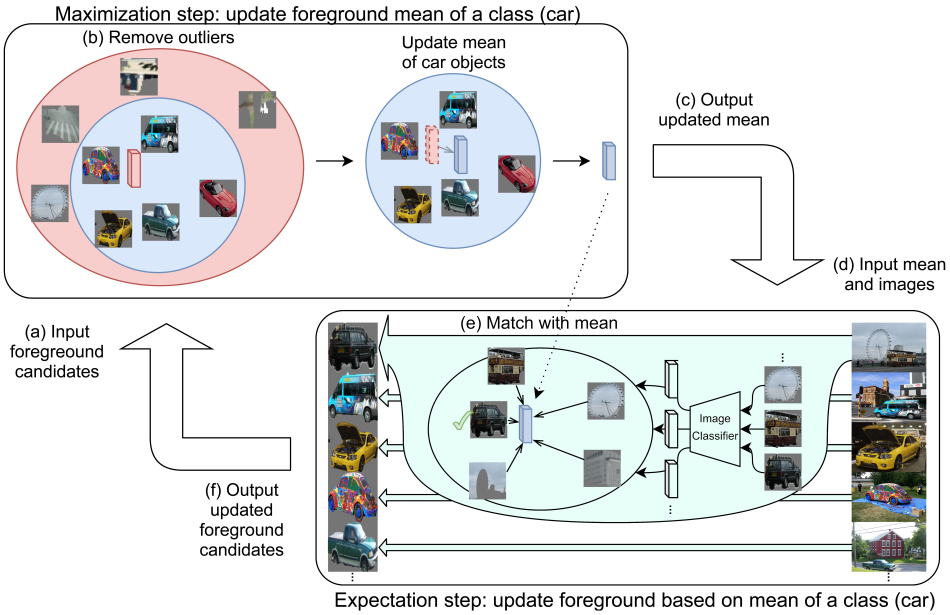


Figure 2: Step 2 and 3 of foreground extraction. (a) Each extracted foreground is passed to the classifier, and a latent representation of the image is extracted using a bottleneck layer. (b) Using the mean of all latent representations, we keep k% representations that are close to the mean and rule out outliers. (c) The mean is updated after ruling out the outliers. (d) For each image, latent representations of all eligible segments are obtained by the classifier network. (e) The segment with the highest cosine similarity to the updated mean is selected as the new foreground of the image. (f) After obtaining a new set of foregrounds, they are used as input of step 2 of the next iteration.

In this case, foreground object may not perfectly overlap with the Grad-CAM activation map, because c_i^a is the mean position of multiple foreground objects. To resolve this problem, we keep top-n segments with the shortest distances to the center c_i^a. Typically $n \leq 3$. Then, we use the image classifier as an additional **semantic metric** to select the correct foreground. Specifically, we pass the top-n segments through the same image classifier $f(\cdot)$ used in Grad-CAM. The segment with the highest predicted probability is our initial selection of the foreground of the input image O_i. While, the initial extraction is far from perfect, because grad-cam localize the most discriminative location depends only on high-level classification information, which may have mismatch to the correct object location given complex scene. To resolve the above issues and to further improve foreground object masks, we propose an iterative approach whereby we select a subset of segments $\mathcal{O}$ from the larger set of original segments $\mathcal{S}$ generated by the region proposal method. These selected segments are considered as foreground object segments. We frame the iterative segment selection within an Expectation-Maximization like steps.

The following EM steps assume that the latent representation (through a feature extractor $\phi(\cdot)$ of the image classifier $f(\cdot)$) of all foreground objects from the same class a follow a distribution p_{ψ_a}, here we assume $p_{\psi_a} \sim \mathcal{N}(\mu, \Sigma)$ follows Multivariate Gaussian distribution because in the latent space of image classifier $f(\cdot)$, representations of images from same class should be a single cluster. Our goal is to find the optimal parameters of the distribution. We follow an expectation maximization (EM-) like approach to find optimal parameters. This involves iteratively optimizing the distribution parameter ψ_a which includes $\mu \in \mathbb{R}^d$ and $\Sigma \in \mathbb{R}^{d \times d}$ for each interest class a (Maximization step). Then use ψ_a to find accurate foreground segments in the latent space of $\phi(\cdot)$ (Expectation step). Fig. 2 shows the whole process.

Step 2 (M-step) Maximization. In M-step, for each interest class a, our goal is to find the optimal parameters μ_a, Σ_a given the candidate foreground proposals $\mathcal{O}$. Here $\mathcal{O}$ are extracted foreground objects (from step 1, or step 3 at the previous iteration) for a specific class a. For each segment O_i, we generate its latent space representation h_i by passing it through the image classifier $h_i = \phi(O_i)$. In particular, h_i is the feature after the last convolution layer of the classifier. We compute $\hat{\mu} = \mathbb{E}(\psi) = \frac{1}{N}\sum_{i=1}^{N} h_i$ and $\hat{\Sigma} = \mathbb{E}((h-\hat{\mu})(h-\hat{\mu})^T)$ as initial mean vector and covariance matrix of latent space representations of all selected foreground segments. Because not all foreground masks O_i are correct foregrounds, some of them may be background objects or objects with a different label. To remove the outlier and update the mean vector, we rule out outliers by keeping only $k\%$ of the segments that

are closest to $\hat{\mu}$ based on the Mahalanobis distance m-dist$(h_i, (\hat{\mu}, \hat{\Sigma})) = \sqrt{(h_i - \hat{\mu})^T \hat{\Sigma}^{-1} (h_i - \hat{\mu})}$ of their latent representations. Using only the remaining foreground (inlier) segments, we compute a new mean $\hat{\mu}'$ and covariance matrix $\hat{\Sigma}'$ of the foreground object latent representations of the given class, which can be used to match more accurate foreground mask in E-step.

Algorithm 1 F-EM

Input: Set of images $\mathcal{I} = \{I_i\}_{i=1}^N$, set of labels $\mathcal{Y} = \{\mathbf{y_i}\}_{i=1}^N$, image classifier $f(\cdot)$ with feature extractor $\phi(\cdot)$, class of interest a

Output: Set of foregrounds $\mathcal{O} = \{O_i\}_{i=1}^N$ of class a objects.

$\mathcal{O} \leftarrow \emptyset$ ▷ Step 1
for all image $I_i \in \mathcal{I}$ where $y_i^a = 1$ **do**
  $\mathcal{S}_i \leftarrow$ EntitySeg(I_i)
  $c_i^a \leftarrow$ center of Grad-CAM(I_i, a)
  $p_s \leftarrow f(s)$ for n of $s \in \mathcal{S}_i$ with smallest dist(s, c_i^a)
  $\mathcal{O} \leftarrow \mathcal{O} \cup \{\arg\max_s \{p_s\}\}$
for j iterations **do** ▷ Step 2 (Maximization)
  $\hat{\mu}, \hat{\Sigma} \leftarrow$ mean and covar of $\{\phi(O_i)\}_{i=1}^N$
  $\mathcal{O}' \leftarrow$ k% of $O_i \in \mathcal{O}$ with smallest m-dist$(\phi(O_i), (\hat{\mu}, \hat{\Sigma}))$
  $\hat{\mu}', \hat{\Sigma}' \leftarrow$ mean and covar of $\{\phi(O_i')\}_{i=1}^{N \times k\%}$
  $\mathcal{O} \leftarrow \emptyset$ ▷ Step 3 (Expectation)
  for all image $I_i \in \mathcal{I}$ where $y_i^a = 1$ **do**
    $\mathcal{S}_i \leftarrow$ EntitySeg(I_i)
    $\mathcal{O} \leftarrow \mathcal{O} \cup \{\arg\min_{s \in \mathcal{S}_i}$ m-dist$(\phi(s), (\hat{\mu}', \hat{\Sigma}'))\}$

Step 3 (E-step) Expectation. In E-step, we regenerate the set of foreground segments $\mathcal{O}$ of class a by matching segment candidates with the updated $\hat{\mu}'$ and $\hat{\Sigma}'$ in M-step. In other words, we compute the "expectation" of the foreground mask for each image: $\mathbb{E}(O_i|\hat{\mu}', \hat{\Sigma}', I_i)$. For each image I_i, we start with the set of all eligible segments $\mathcal{S}_i$ (computed in step 1) again and generate the corresponding latent space representations $\Phi_i = \{h_i^1, h_i^2, ..., h_i^m\}$ as described earlier. We then compute a Mahalanobis distance (m-dist) between each latent representation of the segment h_i^j and the new mean $\hat{\mu}'$ obtained from step 2. The segment with the smallest m-dist is selected as the new foreground of the image. With a new set of foreground segments $\mathcal{O}$, we can perform step 2 followed by step 3 again for several, typically 2 or 3, iterations.

3.2 Background (Context) Augmentation

Next step involves generating a large set of high-quality context images that could be used as background images for pasting foreground masks. One possible approach would be to use randomly selected web images as background images. However, prior works (Dvornik et al., 2018; Divvala et al., 2009; Yun et al., 2021) have shown that context affects model's capacity for object recognition. Thus, selecting appropriate context images is important for learning good object representation, and thus beneficial for instance segmentation as well. To this end, we propose to use image captioning followed by text-to-image image synthesis to automatically generate background images that could provide good contextual information. More details in appendix.

Image Captioning Given an training set image, we leverage an off-the-shelf self-critique sequence training image captioning method (Rennie et al., 2017) to describe the image, but we note that our method is agnostic to any specific image captioning method. These descriptions can capture the important context information. We further design a simple rule to substitute the object words, that has overlap with target interest class (in VOC or COCO) with other object words, in captions, to decrease the possibility of generating images that contains interest object, since they come without labels.

Image Synthesis We use the captions as inputs to text-to-image synthesis pipeline DALL·E (Ramesh et al., 2021)[1], to synthesize a large set of high-quality images that capture all relevant contextual information for performing paste operation (§3.3). For each caption, we generate five synthesized images. Note that with our caption pruning rule described above, we assume that synthesized images do not contain foreground objects.

3.3 Compositional Paste

After foreground extraction (§3.1), we have a pool of extracted foregrounds where each class has a set of corresponding foreground objects. After background augmentation (§3.2) we have both the original background images and contextual augmented background images by DALL·E. We can create a synthetic dataset with pseudo instance segmentation labels by pasting the foreground masks onto the background images. For each background image, we select n_p foregrounds based on a

[1] In implementation, we use Ru-DALL·E (sbe).

(a) The original image to be pasted.

(b) The **red circle** is the max inscribing circle found based on contour, denoted by the **blue line**.

(c) The first object, a person, is pasted on this image.

(d) After repeatedly applying above steps, four objects are pasted on the image.

Figure 3: Illustrative example of Space Maximize Paste algorithm. In this example, four foreground objects are pasted on the background image that contains an aeroplane. In part (b) the max inscribing circle is found from contour based on region without aeroplane. We emphasize that the contour is found only based on image level, using process described in §3.1. Note that the person is scaled to match the size of the circle found in part (b), and a random rotation is performed.

pre-defined distribution p, discussed later, and the goal is to paste those extracted foregrounds with the appropriate size. The appropriate choice of n_p depends on the dataset. To force the model to learn a more robust understanding, each pasted foreground undergoes a random 2D rotation and a scaling augmentation. In addition, we note that direct object pasting might lead to unwanted artifacts, also shown in the findings of Dwibedi et al. (2017) and Ghiasi et al. (2021). To mitigate this issue, we apply a variety of blendings on each pasted image, including Poisson blurring, Gaussian blurring, no blending at all, or any combination of those. In practice, we find Gaussian blurring alone can yield sufficiently strong performance. Now we present two methods, each with their edges, of how to find the paste location. We leave the end-user to decide which method to use.

Random Paste In this simple method, we iteratively scale the foreground object by a random factor $\sim \text{Uniform}(0.3, 1.0)$, and paste in a random location on the image. We find a factor > 1 generally creates objects too large and a small factor enhances model learning capacity for small objects.

Space Maximize Paste This dynamic pasting algorithm tries to iteratively utilize the remaining available background regions to paste foreground objects. Our intuition is to force the pasted foregrounds to occupy as many spaces of the pasted background as possible, while remaining non-overlapping with the new to-be-pasted foreground and original background plus already pasted foregrounds. We give an illustrative example in Fig. 3. Firstly, we find background regions where no object lies by computing the maximum inscribing circle from the contour of background images without existing foregrounds, original or pasted, as shown in the **red circle** in Fig. 3b. The maximum inscribing circle gives a maximum region not occupied by any objects, thus providing the largest empty space. Next, we scale the pasted foreground to largely match the size of the radius of the maximum inscribing circle, rotate by a random degree, and paste to the location of the center of the maximum inscribing circle, shown in Fig. 3c. We iteratively repeat the above steps to paste all n_p foregrounds (Fig. Fig. 3d). We note that since this method finds the background space with the decreasing area, thus able to synthesize images pasted with objects of various sizes.

Selection Probability The pre-defined selection distribution p is crucial in that it imposes the class distribution of synthetic dataset produced by the paste method. We investigate and provide two types of probability to end users. The simplest type is a uniform distribution, i.e., selecting each image from foreground pool with the same chance. With this choice, the synthetic data approximately follows the class distribution of foreground pool. The second type is a balanced sampling, i.e. giving the classes with more instances a smaller weight to be selected while giving the classes with less instances a larger weight. This type enforces each class to appear in synthetic data in approximately the same quantity. In §4.4 we show that this setting is beneficial for long-tail problem.

4 Experiments

We demonstrate the effectiveness of Exact in weakly-supervised instance segmentation from image-level labels on Pascal VOC and MS COCO datasets. Additionally, we also show that Exact is generalizable to object detection task and highlight benefits of Exact in handling long-tail class distribution with only image-level label information.

4.1 Experiment Setup

Dataset and Metrics We evaluate Exact on Pascal VOC (Everingham et al., 2010) and MS COCO (Lin et al., 2014b) datasets. Pascal VOC consists of 20 foreground classes. Further, following common practice of prior works (Ahn et al., 2019; Arun et al., 2020; Sun et al., 2020), we use the augmented

Table 1: Metrics for instance segmentation models on Pascal VOC 2012 val set. Here $\mathcal{F}$ means fully supervised, $\mathcal{B}$ and $\mathcal{I}$ mean bounding box and image level label based weakly supervised approaches respectively. We highlight the best mAP with image level label in green, and bounding box label in blue. Note that our method outperforms prior SOTA image level methods. Further our method achieves better performance than some of the prior bounding box SOTA, although bounding box method has access to a lot more information about object instances.

Method	Supervision	Backbone	$\text{mAP}_{0.50}$	$\text{mAP}_{0.75}$
Mask R-CNN (He et al., 2017)	$\mathcal{F}$	R-101	67.9	44.9
SDI (Khoreva et al., 2017)	$\mathcal{B}$	R-101	44.8	46.7
Liao et al. (2019)	$\mathcal{B}$	R-50	51.3	22.4
Sun et al. (2020)	$\mathcal{B}$	R-50	56.9	21.4
ACI Arun et al. (2020)	$\mathcal{B}$	R-101	58.2	32.1
BBTP (Hsu et al., 2019)	$\mathcal{B}$	R-101	58.9	21.6
PRM (Zhou et al., 2018)	$\mathcal{I}$	R-50	26.8	9.0
IAM (Zhu et al., 2019)	$\mathcal{I}$	R-50	28.8	11.9
OCIS (Cholakkal et al., 2019)	$\mathcal{I}$	R-50	30.2	14.4
Label-PEnet (Ge et al., 2019)	$\mathcal{I}$	R-50	30.2	12.9
CL (Hwang et al., 2021)	$\mathcal{I}$	R-50	38.1	12.3
WISE (Laradji et al., 2019)	$\mathcal{I}$	R-50	41.7	23.7
BESTIE (Kim et al., 2021)	$\mathcal{I}$	R-50	41.8	24.2
JTSM (Shen et al., 2021a)	$\mathcal{I}$	R-18	44.2	12.0
IRN (Ahn et al., 2019)	$\mathcal{I}$	R-50	46.7	23.5
LLID (Liu et al., 2020)	$\mathcal{I}$	R-50	48.4	24.9
PDSL (Shen et al., 2021b)	$\mathcal{I}$	R-101	49.7	13.1
ACI (Arun et al., 2020)	$\mathcal{I}$	R-50	50.9	28.5
BESTIE + Refinement (Kim et al., 2021)	$\mathcal{I}$	R-50	51.0	26.6
EXACT (Ours)	$\mathcal{I}$	R-50	56.2	35.5
EXACT (Ours)	$\mathcal{I}$	R-101	58.4	37.2

version (Hariharan et al., 2011) with 10,582 training images, and 1,449 val images for Pascal VOC dataset. MS COCO dataset consists of 80 foreground classes with 118,287 training and 5,000 test images. Per the standard instance segmentation and object detection evaluation protocol, we report mean average precision (mAP) (Hariharan et al., 2014) on two different intersection-over-union (IoU) thresholds, namely, 0.5 and 0.75. We denote these two mAPs as $\text{mAP}_{0.50}$ and $\text{mAP}_{0.75}$, respectively.

Synthesized Training Dataset We do not touch on the segmentation label but instead generate pseudo-labeled synthesized training dataset using methods described in Sec. §3. Pascal VOC training dataset of 10,582 images consists of 29,723 objects in total, we extract 10,113 masks of foreground segments (34.0%). Similarly, MS COCO training set of 118,287 images consists of 860,001 objects in total, we extract 192,731 masks of foreground segments (22.4%) [2]. We observe that such masks are not perfect and contain noise, but overall have sufficient quality. To further ensure the quality of foregrounds, we filter the final results using 0.1 classifier score threshold. Additionally, we leverage image captioning and DALL·E (§3.2) to further contextually augment backgrounds. We generate 2 captions per image[3], synthesize 10 contextual backgrounds per caption, and utilize CLIP (Radford et al., 2021) to select top 5 backgrounds among the 10 synthesized images, together producing 10k and 118k augmented backgrounds for VOC and COCO respectively. To make the best use of both original backgrounds and contextually augmented backgrounds, we blend them together as our background pool, on which we apply methods from §3.3 to paste these extracted foregrounds. For simplicity, we use Random Paste method. We duplicate the original backgrounds twice to make the distribution between real and synthetic backgrounds more balanced.

Model Architecture and Training Details We train Mask R-CNN (He et al., 2017) with Resnet 50 (R-50) or Resnet 101 (R-101) as backbone (He et al., 2016). We initialize Resnet from ImageNet (Deng et al., 2009) pretrained weight released by `detectron2` (Wu et al., 2019) We deploy large-scale jittering (Ghiasi et al., 2021), and additionally augment training data with random brightness and random contrast with probability 0.5. We run our experiments on one 32GB Tesla V100 GPU with learning rate of 0.1 and batch size of 128.

4.2 WEAKLY-SUPERVISED INSTANCE SEGMENTATION

In our setting, we follow the details in §4.1 and assume access to only image-level labels. That is, we *do not* use any segmentation annotation from the training set. We report VOC performance in Tab. 1.

[2]We provide the number of per-class foreground masks extracted in appendix.

[3]We augment each of 10,582 VOC image, and a random 10% sample of 118k COCO images.

Baselines We compare against previous weakly-supervised SOTA. Notably, Zhou et al. (2018) utilize peak response maps from image-level multi-label object classification model to infer instance mask from pre-computed proposal gallery; Laradji et al. (2019) generate pseudo-label training set from MCG (Arbeláez et al., 2014) and train a supervised Mask-RCNN (He et al., 2017); and Khoreva et al. (2017) generate pseudo-label using `GradCut+` and MCG (Arbeláez et al., 2014).

Table 2: Weakly supervised instance segmentation on MS COCO val2017. Models reported here use image level label.

Method	Backbone	$\text{mAP}_{0.50}$	$\text{mAP}_{0.75}$
WS-JDS (Shen et al., 2019)	VGG16	11.7	5.5
JTSM (Shen et al., 2021a)	R-18	12.1	5.0
PDSL (Shen et al., 2021b)	R-18	13.1	5.0
IISI (Fan et al., 2018)	R-101	25.5	13.5
LLID (Liu et al., 2020)	R-50	27.1	16.5
BESTIE (Kim et al., 2021)	R-50	28.0	13.2
EXACT (Ours)	R-50	**30.8**	**20.7**

Results Quantitative results on Pascal VOC dataset have been shown in Tab. 1. Firstly, we experiment with the choice of R-50 or R-101. With more capacity brought by a deeper model, we find that R-101 works better compared to R-50, leading to +2.2 $\text{mAP}_{0.50}$ and +1.7 $\text{mAP}_{0.75}$ improvement. This validates that EXACT is suitable for instance segmentation task. Secondly, our method can significantly outperform previous image-level SOTA by +7.4 $\text{mAP}_{0.50}$ (from 51.0 to 58.4), and +8.7 $\text{mAP}_{0.75}$ improvements (from 28.5 to 37.2). This suggests that pseudo-labels generated by EXACT give a strong learning signal for the model to develop object awareness. Lastly, although bounding box is a more insightful cure for instance segmentation, we find our results comparable with the previous SOTAs that use bounding box. Indeed, we are only 0.5 $\text{mAP}_{0.50}$ lower compared to the best bounding box SOTA, which requires hand-drawn ground-truth boxes.

Next we demonstrate effectiveness of the proposed EXACT method on MS COCO dataset (Lin et al., 2014b). It is much more challenging than the Pascal VOC dataset as it consists of 80 object classes and each image may contain multiple instances of different classes. Quantitative results are shown in Tab. 2. We observe that the proposed method can achieve an improvement of +2.8 $\text{mAP}_{0.50}$, and +7.5 $\text{mAP}_{0.75}$ improvements over previous image-level SOTA. Interestingly, our method with smaller architecture (R-50) outperform prior method IISI (Fan et al., 2018) that works with larger network (R-101). These results provide evidence that our method can scale to large data with large number of object classes.

Table 3: Ablation study on PASCAL VOC.

DALL·E	# Paste Objects	Foreground	$\text{mAP}_{0.50}$	$\text{mAP}_{0.75}$
✗	4	-	56.5	35.8
Black	4	-	55.7	34.3
Random	4	-	56.3	36.7
✓	4	w/o Alg. 1	53.4	35.7
✓	2	Balanced Selection	56.8	36.3
✓	2	-	56.9	36.6
✓	$1 \sim 4$	-	58.0	**38.1**
✓	6	-	57.2	37.5
✓	4	-	**58.4**	37.2

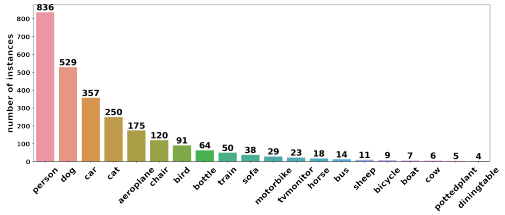
Ablation Study We present the performance of EXACT on PASCAL VOC 2012 val set with different choice of parameters in Tab. 3. All experiments use R-101 as backbone, training with synthetic data generated by Sec. §3 and following the details in §4.1. We first note that DALL·E is indispensable for best performance, and training only with 20,226 backgrounds from original background pool gives 1.9 lower $\text{mAP}_{0.50}$, validating our hypothesis that additional contextual background makes model learns more thorough object representation. Furthermore, it is crucial to choose an appropriate backgrounds. A purely black background or a random background[4] does not bring benefit but in turn harm model learning (0.8 and 0.2 $\text{mAP}_{0.50}$ lower than not using additional augmented images). Additionally, we quantify the effect of Alg. 1 by training a model on foreground extracted without F-EM, and observe that iterative foreground refinement is essential for a quality foreground, as F-EM provides 5.0 $\text{mAP}_{0.50}$ improvement. Moreover, for the original PASCAL VOC dataset, balanced selection might not work particularly well overall, giving 1.6 $\text{mAP}_{0.50}$ lower. Given 30k training set, a balanced selection makes each class approximately 1.5k, and classes with a smaller set of extracted foreground objects will be reused more often, and the potential noise from extraction in §3.1 might be amplified. This result suggests a more sophisticated selection method is needed, which we leave for future work. Lastly, the number of paste objects n_p is important in that a value too low results in sparse foregrounds, while a value

Table 4: Object detection on Pascal VOC 2012.

Method	Backbone	$\text{mAP}_{0.50}$	$\text{mAP}_{0.75}$
Wetectron (Ren et al., 2020)	VGG16	52.1	-
CASD (Huang et al., 2020)	VGG16	53.6	-
EXACT (Ours)	R-50	57.2	30.7

[4]For simplicity we use MS COCO images that does not contain any of 20 VOC objects as random background.

(a) Long-tail distribution of generated data (Wu et al., 2020). The number of instances for each class shown on the top.



(b) We report mAP@50 for each class. **Gray** values are from Mask RCNN trained directly on data with extracted mask (§3.1); values in **red** are the value after EXACT. The classes are ordered the same as (a).

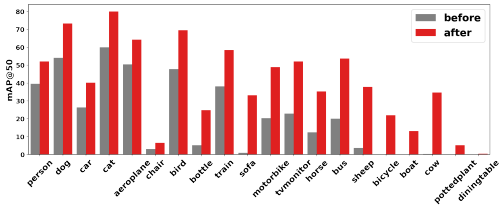


Figure 4: Long-tail instance segmentation setting and results.

too large results in crowded foregrounds, each of those hurts the model learning. We empirically show that for PASCAL VOC dataset, 4 seems to be a more appropriate value to use. Surprisingly a fixed 4 gives slightly higher $\text{mAP}_{0.50}$ compared to assigning random $n_p \sim \text{Unif}[1, 4]$ dynamically. More analysis on region proposal methods and fore-ground mask quality are in Appendix.

4.3 Weakly-supervised Object Detection

We argue that EXACT is effective not only in instance segmentation task, but on other tasks as well. We reuse synthesized dataset described in §4.1 to conduct object detection on Pascal VOC dataset. We compare our method against two popular baseline methods, CASD Huang et al. (2020) and Wetectron Ren et al. (2020). In Tab. 4, we observe that our method can achieve almost +4.0 and +5.0 $\text{mAP}_{0.50}$ compared to CASD and Wetectron respectively.

4.4 Weakly-supervised Instance Segmentation on Long-tail Dataset

We now discuss how EXACT can alleviate the long tail problem. In long-tail (He & Garcia, 2009) dataset, the head class objects contain much more instances compared to tail class objects, so that simple learning method might learn the bias of the dataset, that is, become the majority voter which predicts the head class all the time (Liu et al., 2019; Wang et al., 2021). Due to the ability to generate synthetic data based on selection distribution, even given a highly imbalanced dataset, we can create a synthetic dataset with a balanced class distribution.

Implementation Detail We conduct our experiments on a long-tailed version of PASCAL VOC (Everingham et al., 2010) dataset. Our long-tailed dataset, generated based on method proposed in Wu et al. (2020), forces the distribution of each class to follow Pareto distribution (DAVIS & FELDSTEIN, 1979). It contains 2,415 images in total, with a maximum of 836 and a minimum of 4 masks in a class. Statistics of our generated dataset shown in Fig. 4a. The `person` class contains the most instances, while there are 5 classes with less than 10 instances. To the best of our knowledge, we are the first to conduct weakly-supervised instance segmentation task using Wu et al. (2020).

Results We now show that our weakly-supervised instance segmentation can largely mitigate the long tail problem. Our results on PASCAL VOC val set are shown in Fig. 4b, with $\text{mAP}_{0.50}$ values for each class. We compare with Mask RCNN (He et al., 2017) with details described in §4.1, using the long tail dataset itself, i.e. only train with pseudo-labels inferred by §3.1. As shown in Fig. 4a, training on such an imbalanced data deteriorates the model. Out of 20 classes, there are 10 classes that have $\text{mAP}_{0.50} \leq 12.42$, 6 classes that have $\text{mAP}_{0.50} < 1$ and 4 classes that are not being recognized by model at all. The overall $\text{mAP}_{0.50}$ is 20.26. However, after EXACT with balanced setting (the p in §3.3), the model increasing overall $\text{mAP}_{0.50}$ to 40.28. All classes show an improvement compared to vanilla training, with an average improvement of 20.0 $\text{mAP}_{0.50}$.

5 Conclusion

We propose EXACT: EXtract-AugContext-pasTe, a compositional image augmentation pipeline for weakly supervised instance segmentation method using only image-level supervision. The core of our approach involves proposing an EM-like iterative method for foreground object mask generation and then compositing them on context-aware background images. We demonstrate the effectiveness of our approach on the Pascal VOC 2012 instance segmentation task by using only image-level labels. Our method significantly outperforms the best baselines. Further, the method also achieves state-of-the-art accuracy on the long-tail weakly-supervised instance segmentation problem.

REFERENCES

sberbank-ai/ru-dalle: Generate images from texts. in russian. https://github.com/sberbank-ai/ru-dalle. (Accessed on 03/07/2022).

Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.

Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2209–2218, 2019.

Pablo Arbeláez, Jordi Pont-Tuset, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 328–335, 2014.

Aditya Arun, CV Jawahar, and M Pawan Kumar. Weakly supervised instance segmentation by learning annotation consistent instances. In *European Conference on Computer Vision*, pp. 254–270. Springer, 2020.

G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

Hisham Cholakkal, Guolei Sun, Fahad Shahbaz Khan, and Ling Shao. Object counting and instance segmentation with image-level supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12397–12405, 2019.

HENRY T. DAVIS and MICHAEL L. FELDSTEIN. The generalized pareto law as a model for progressively censored survival data. *Biometrika*, 66(2):299–306, 1979. doi: 10.1093/biomet/66.2.299. URL https://doi.org/10.1093/biomet/66.2.299.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. Cogview: Mastering text-to-image generation via transformers. *Advances in Neural Information Processing Systems*, 34, 2021.

Santosh K Divvala, Derek Hoiem, James H Hays, Alexei A Efros, and Martial Hebert. An empirical study of context in object detection. In *2009 IEEE Conference on computer vision and Pattern Recognition*, pp. 1271–1278. IEEE, 2009.

Nikita Dvornik, Julien Mairal, and Cordelia Schmid. Modeling visual context is key to augmenting object detection datasets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 364–380, 2018.

Debidatta Dwibedi, Ishan Misra, and Martial Hebert. Cut, paste and learn: Surprisingly easy synthesis for instance detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 1301–1310, 2017.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

Ruochen Fan, Qibin Hou, Ming-Ming Cheng, Gang Yu, Ralph R Martin, and Shi-Min Hu. Associating inter-image salient instances for weakly supervised semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 367–383, 2018.

Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181, 2004.

Weifeng Ge, Sheng Guo, Weilin Huang, and Matthew R Scott. Label-penet: Sequential label propagation and enhancement networks for weakly supervised instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3345–3354, 2019.

Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2918–2928, 2021.

Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5356–5364, 2019.

Abdul Mueed Hafiz and Ghulam Mohiuddin Bhat. A survey on instance segmentation: state of the art. *International journal of multimedia information retrieval*, 9(3):171–189, 2020.

Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 international conference on computer vision*, pp. 991–998. IEEE, 2011.

Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *European conference on computer vision*, pp. 297–312. Springer, 2014.

Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

Tomáš Hodaň, Vibhav Vineet, Ran Gal, Emanuel Shalev, Jon Hanzelka, Treb Connell, Pedro Urbina, Sudipta N Sinha, and Brian Guenter. Photorealistic image synthesis for object instance detection. In *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 66–70. IEEE, 2019.

Ting-I Hsieh, Esther Robb, Hwann-Tzong Chen, and Jia-Bin Huang. Droploss for long-tail instance segmentation. In *AAAI*, volume 3, pp. 15, 2021.

Cheng-Chun Hsu, Kuang-Jui Hsu, Chung-Chi Tsai, Yen-Yu Lin, and Yung-Yu Chuang. Weakly supervised instance segmentation using the bounding box tightness prior. *Advances in Neural Information Processing Systems*, 32, 2019.

Yuan-Ting Hu, Hong-Shuo Chen, Kexin Hui, Jia-Bin Huang, and Alexander G Schwing. Sail-vos: Semantic amodal instance level video object segmentation-a synthetic dataset and baselines. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3105–3115, 2019.

Z. Huang, Y. Zou, B. V. K. Vijaya Kumar, and D. Huang. Comprehensive attention self-distillation for weakly-supervised object detection. In *Neural Information Processing Systems (NeurIPS)*, 2020.

Jaedong Hwang, Seohyun Kim, Jeany Son, and Bohyung Han. Weakly supervised instance segmentation by deep community learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1020–1029, 2021.

Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 876–885, 2017.

Beomyoung Kim, Youngjoon Yoo, Chaeeun Rhee, and Junmo Kim. Beyond semantic to instance segmentation: Weakly-supervised instance segmentation via semantic knowledge transfer and self-refinement. *arXiv preprint arXiv:2109.09477*, 2021.

Issam H Laradji, David Vazquez, and Mark Schmidt. Where are the masks: Instance segmentation with image-level supervision. *arXiv preprint arXiv:1907.01430*, 2019.

Shisha Liao, Yongqing Sun, Chenqiang Gao, Pranav Shenoy KP, Song Mu, Jun Shimamura, and Atsushi Sagata. Weakly supervised instance segmentation using hybrid networks. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1917–1921. IEEE, 2019.

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, 2014a. URL https://arxiv.org/abs/1405.0312.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014b.

Yun Liu, Yu-Huan Wu, Pei-Song Wen, Yu-Jun Shi, Yu Qiu, and Ming-Ming Cheng. Leveraging instance-, image-and dataset-level information for weakly supervised instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2537–2546, 2019.

Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens Van Der Maaten. Exploring the limits of weakly supervised pretraining. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 181–196, 2018.

K.K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool. Convolutional oriented boundaries. In *European Conference on Computer Vision (ECCV)*, 2016.

Lu Qi, Jason Kuen, Yi Wang, Jiuxiang Gu, Hengshuang Zhao, Zhe Lin, Philip Torr, and Jiaya Jia. Open-world entity segmentation. *arXiv preprint arXiv:2107.14228*, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831. PMLR, 2021.

Z. Ren, Z. Yu, X. Yang, M. Liu, Y. J. Lee, A. G. Schwing, and J. Kautz. Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7008–7024, 2017.

Stephan R Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2213–2222, 2017.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Yunhang Shen, Rongrong Ji, Yan Wang, Yongjian Wu, and Liujuan Cao. Cyclic guidance for weakly supervised joint detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 697–707, 2019.

Yunhang Shen, Liujuan Cao, Zhiwei Chen, Feihong Lian, Baochang Zhang, Chi Su, Yongjian Wu, Feiyue Huang, and Rongrong Ji. Toward joint thing-and-stuff mining for weakly supervised panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16694–16705, 2021a.

Yunhang Shen, Liujuan Cao, Zhiwei Chen, Baochang Zhang, Chi Su, Yongjian Wu, Feiyue Huang, and Rongrong Ji. Parallel detection-and-segmentation learning for weakly supervised instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8198–8208, 2021b.

Yongqing Sun, Shisha Liao, Chenqiang Gao, Chengjuan Xie, Feng Yang, Yue Zhao, and Atsushi Sagata. Weakly supervised instance segmentation based on two-stage transfer learning. *IEEE Access*, 8:24135–24144, 2020.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8769–8778, 2018.

Olga Veksler, Yuri Boykov, and Paria Mehrani. Superpixels and supervoxels in an energy optimization framework. In *European conference on Computer vision*, pp. 211–224. Springer, 2010.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Jiaqi Wang, Wenwei Zhang, Yuhang Zang, Yuhang Cao, Jiangmiao Pang, Tao Gong, Kai Chen, Ziwei Liu, Chen Change Loy, and Dahua Lin. Seesaw loss for long-tailed instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9695–9704, 2021.

Tong Wu, Qingqiu Huang, Ziwei Liu, Yu Wang, and Dahua Lin. Distribution-balanced loss for multi-label classification in long-tailed datasets. In *European Conference on Computer Vision (ECCV)*, 2020.

Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

Woo-Han Yun, Taewoo Kim, Jaeyeon Lee, Jaehong Kim, and Junmo Kim. Cut-and-paste dataset generation for balancing domain gaps in object instance detection. *IEEE Access*, 9:14319–14329, 2021.

Yanzhao Zhou, Yi Zhu, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. Weakly supervised instance segmentation using class peak response. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3791–3800, 2018.

Yi Zhu, Yanzhao Zhou, Huijuan Xu, Qixiang Ye, David Doermann, and Jianbin Jiao. Learning instance activation maps for weakly supervised instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3116–3125, 2019.

APPENDIX

A DETAILS OF FOREGROUND EXTRACTION

In this section we provide more details regarding the first step of EXACT: foreground extraction. We also give the number of per-class foreground masks extracted in Tab. 5.

First step in our pipeline is the generation of a set of object proposal using region proposal methods. In general from the set of generated object region proposals, we ignore segments with very small area (<1% image size), as well as segments whose bounding boxes are almost same as the size of the whole image in at least one dimension. We observe that most of such segments are noise or background segments.

In order to get feature for each segment, for each segment O_i, we first use erosion algorithm (Bradski, 2000) to remove small potential artifacts and noisy pixels in the segment. Then, we paste the foreground segment onto a minimum bounding box resized to a square with gray padding before passing it to the classifier.

We give five examples demonstrating our first step in Fig. 5. The first row is the original image. Each image in row 2 is segmented by entity segmentation (Qi et al., 2021). Note that this method is generic and does not require instance level label. Row 3 is the Grad-CAM (Selvaraju et al., 2017) outcome for target image label, where the heatmap represents model's attention over the image. Our method which combines the best of entity segmentation and Grad-CAM, is able to fetch accurate foreground masks of objects. For our EM algorithm, a sample visual output of two iterations is shown in Fig. 6. Our method is able to further rule out background segments and refine foreground selections through a few EM iterations.

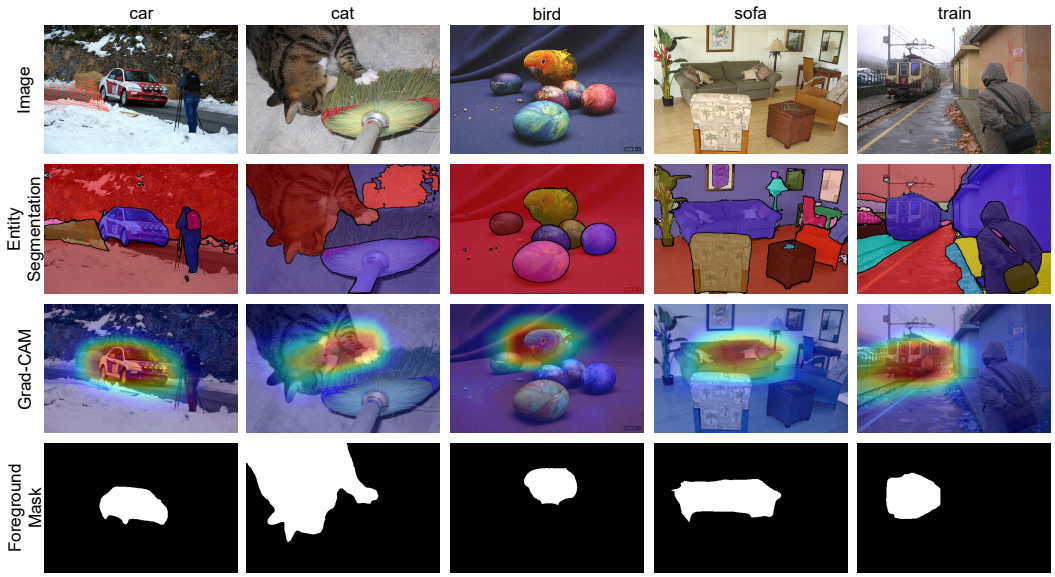


Figure 5: Five sample images and their results of entity segmentation (row 2), Grad-CAM (row 3), and foreground mask selected (row 4, after step 1 of EXACT).

B DETAILS OF BACKGROUND (CONTEXT) AUGMENTATION

We provide the detail of how DALL·E is used in background augmentation step of EXACT. We provide an illustration in Fig. 7. Given an input image alone, using image captioning method (Rennie et al., 2017), we can obtain a caption that describes the image and its context. We observe that even if the caption is rough, as in the example in Fig. 7, DALL·E is still able to generate good context.

However, there is a caveat to use those captions directly: when the caption contains an object, say, "person", the generated images may also contain "person". If "person" is generated, such generated

Figure 6: Sample visual output of EM iterations on foregrounds selected from bus category. The box on the left shows the result after step 1 of EXACT as the input of EM algorithm, and the boxes in the middle and right show refined foreground selections in the first and second iterations of the EM algorithm.

aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow
474	229	338	384	331	274	685	790	335	130
dining table	dog	horse	motorbike	person	potted plant	sheep	sofa	train	tv monitor
126	787	280	335	3142	253	198	254	327	441

Table 5: Number of foreground masks extracted per-class in PASCAL VOC dataset.

object naturally does not come with segmentation label, and will mislead models during training. Therefore, for simplicity, we devise a rule to manually replace *any* words related to PASCAL VOC (Everingham et al., 2010) or MS COCO (Lin et al., 2014b) object labels or their synonyms from captions, e.g. "human", "person" and "people" to some distractor words based on supercateogry. Both VOC and COCO release their official mapping of each label to one supercategory. We design several distractor words for each of the 12 supercategory, shown in Tab. 6. For each occurrence of word that is related to one supercategory in captions, we replace such word with a random word from corresponding distractor words. In the example of Fig. 7, the object mention **train**, belong to `Vehicle` supercategory, is replaced by **tank**.

Supercategory	Distractor words
Accessory	coat, T-shirt, gloves, scarf, trousers
Animal	lion, tiger, cheetah, pig, horse, rabbit
Appliance	washing machine, injector, lamp, air conditioner, humidifier
Electronic	battery, computer charger, circuit board, mp4 player, headphone
Food	berry, mango, pineapple, watermelon, olive
Furniture	bookcase, piano, wardrobe, washstand, sofa
Indoor	board game, detergent, coffee, pencil case, drinking straw
Kitchen	water-tap, juicer, sink, hot pot, kettle
Person	monkey, gorilla
Outdoor	tree, billboard, gate
Sports	basketball, tent, ballon, swimming ring, football
Vehicle	tank, helicopter, bulldozer, UFO, scooter

Table 6: Distractor words designed for each supercategory.

Lastly, we use DALL·E to generate five synthesized images for each input image. In this work, we leverage an open-source DALL·E-like implementation Ru-DALL·E (sbe) to generate images.

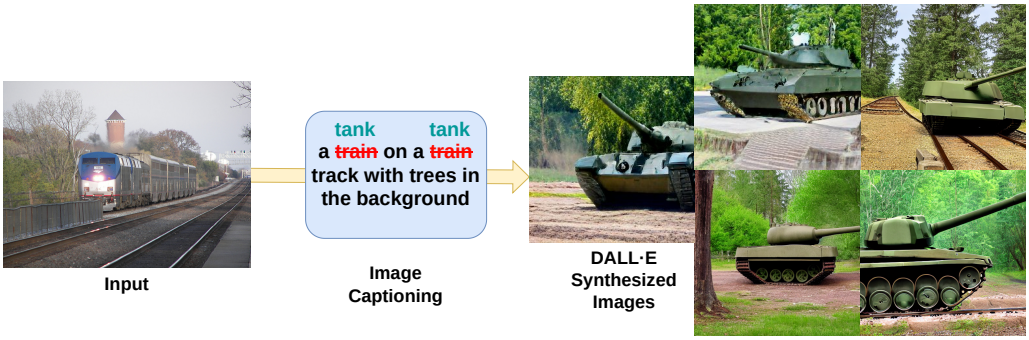


Figure 7: An illustration of background augmentation step of EXACT. Given an input image, we use image captioning to generate caption that describes the input image. Then, we replace any object-related words that overlap with target class from the caption. In this case, a object-related word **train** which exist in the target label of PASCAL VOC occurs twice and such word is replaced by **tank**. Then, we use DALL·E to generate five synthesized images based on the replaced caption.

C PASTE METHOD

In this section, we provide the detailed pseudo-code of **Space Maximize Paste** algorithm described in §3.3. This algorithm is implemented using `opencv` (Bradski, 2000).

Algorithm 2 Space Maximize Paste

Input: images $\mathcal{I} = \{(I_i, \mathcal{F}_i)\}_{i=1}^{N}$, each image I_i has extracted foregrounds $\mathcal{F}_i$; number of paste foregrounds n_p; rotation degree θ; selection distribution p

Output: pasted images $\mathcal{I}$, n_p foregrounds are pasted on each pasted image

for $I_i \in \mathcal{I}$ **do**
  Sample n_p images $\mathcal{I}_p$ from $\mathcal{I}$ based on distribution p
  for $I_p \in \mathcal{I}_p$ **do**
    ▷ paste foregrounds $\mathcal{F}_p$ on I_i
    for foreground $f_p \in \mathcal{F}_p$ **do**
      Find contour $\mathcal{C}$ from region of I_i without $\mathcal{F}_i$
      Calculate maximum inscribing circle from $\mathcal{C}$. Let r_1 be radius and (x, y) be center
      Calculate minimum enclosing circle of f_p. Let r_2 be radius
      Scale and rotate f_p^0 by $\frac{r_1}{r_2}$ and θ', respectively. $\theta' < \theta$ is a random degree.
      Paste f_p on location (x, y) of I_i, $\mathcal{F}_i \leftarrow \mathcal{F}_i \cup \{f_p\}$

We also give a comparison for two paste methods described in §3.3 in Tab. 7. Although Random Paste gives a better overall mAP, we note that it performs relatively poorly in small-size objects, compared to Space Maximize Paste. This is due to Space Maximized Paste displaying all sizes of scaled objects uniformly, and potentially more suitable for datasets with smaller objects.

Table 7: Two paste methods on PASCAL VOC val set. mAP_s, mAP_m and mAP_l represent mAP for small, medium and large objects respectively, with area threshold being 32^2 and 96^2. They are computed as average of IoU range from 0.5 to 0.95 with step 0.05, thus more comprehensive compared to IoU 0.5.

Method	$\text{mAP}_{0.50}$	$\text{mAP}_{0.75}$	mAP_s	mAP_m	mAP_l
Space Maximize	55.1	36.1	**16.3**	29.3	47.0
Random	**58.4**	**37.2**	13.9	**30.7**	**48.0**

Table 8: Two mAP metrics using three kinds of foreground masks. **GT** is ground truth masks on PASCAL VOC vanilla train dataset; **F-EM** is the overlapping extracted foregrounds (see §3.1); **GT** $\cap$ **F-EM** is the overlapping ground truth masks.

Foreground Mask	Use EXACT	Dataset Size	$\text{mAP}_{0.50}$	$\text{mAP}_{0.75}$
GT	✓	1,464 $\rightarrow$ 2928	59.0	37.2
GT	✗	1,464	53.7	21.2
F-EM	✓	1,009 $\rightarrow$ 2,018	44.2	17.0
GT $\cap$ **F-EM**	✓	1,009 $\rightarrow$ 2,018	45.8	16.8

D FURTHER STUDY ON QUALITY OF FOREGROUND MASKS

In this section, we describe preliminary experiments regarding the quality and quantity of foreground masks. We report $\text{mAP}_{0.50}$ and $\text{mAP}_{0.75}$ for three kinds of foreground masks in Tab. 8. All experiments follow the model setup described in §4.1. **GT** represents using ground truth instance level mask as foreground mask. Instance level segmentation masks are not available in the PASCAL VOC trainaug dataset (Hariharan et al., 2011), and we resort to vanilla 1,464 PASCAL VOC train dataset (Everingham et al., 2010) that does come with instance segmentation masks. **F-EM** represents using foreground masks extracted in §3.1. F-EM is conducted on the whole trainaug dataset, among those extracted foregrounds there are 1,009 images overlapping with 1,464 vanilla train dataset. **GT** $\cap$ **F-EM** represents the ground truth masks for these 1,009 overlapping images. For experiments with EXACT enabled, EXACT is applied without the background augmentation step (§3.2). In the column **Dataset Size**, values before the right arrow is the original foreground masks and the values after is the size generated after applying paste algorithm (§3.3). We run paste algorithm (§3.3) twice with number of pasted objects equal to four.

First we investigate how important is the quality of foreground masks. Comparing to row 3 and 4, which are identical except the source of foreground masks used, we observe that ground truth masks did lead to an improvement in $\text{mAP}_{0.50}$, suggesting that accurate foreground mask gives a better cue for object recognition. However, our extracted foreground can give competitive performance. Second we explore the quantity of foreground masks. From row 1 and 4, model benefits from the increasing of the ground truth masks, boosting performance from 45.8 to 59.0 $\text{mAP}_{0.50}$. We credit the performance gains obtained by EXACT partly to the property that EXACT can generated more training sets, thus with significantly more foreground masks. As shown in row 1 and 2, using EXACT will double the dataset size, leading a +5.3 $\text{mAP}_{0.50}$ gain.

Furthermore, we emphasize that EXACT is capable of using any off-the-shelf region proposal methods. In Tab. 9 we observe that using COB (Maninis et al., 2016) still produce comparable performance.

Table 9: EXACT performance on PASCAL VOC val set using different segmentation methods as region proposal in the foreground extraction step.

Method	$\text{mAP}_{0.50}$	$\text{mAP}_{0.75}$
EXACT with COB(Maninis et al., 2016)	56.1	**37.9**
EXACT with entity segmentation (Qi et al., 2021)	**58.4**	37.2

E ERROR STATISTICS

In this section, we present the standard deviations for the errors to highlight the statistical significance of our method improvements. Overall, the improvements are much larger than the standard deviation values.

F LONG TAIL DATASET GENERATION

In this section, we give the detail of how the longtail version of PASCAL VOC dataset (Everingham et al., 2010) is generated using approach proposed in Wu et al. (2020).

Table 10: Metrics for instance segmentation models on Pascal VOC 2012 val set with standard deviations. We also list the backbone architectures.

Method	**Backbone**	**mAP**$_{0.50}$	**mAP**$_{0.75}$
EXACT (Ours)	R-50	56.0 $\pm$ 0.05	38.2 $\pm$ 0.05
EXACT (Ours)	R-101	58.3 $\pm$ 0.03	39.5 $\pm$ 0.03

In Wu et al. (2020), the distribution of each class is imposed to follow Pareto distribution (DAVIS & FELDSTEIN, 1979). Given a shape parameter b which controls the decay rate, the pdf[5] of the distribution is $f(x, b) = \frac{b}{x^{b+1}}$. A rescaled reference discrete distribution of number of data in each class is then generated based on the pdf and the given maximum and minimum number of images among all classes. After sorting the classes in descending order based on the number of available masks, a subset of each class in the dataset is randomly sampled to roughly fit the reference distribution. We generate a dataset based on a Pareto distribution with the shape parameter b equals 6, and the max and min parameters are 800 and 1 respectively.

[5]We leverage `SciPy` library (Virtanen et al., 2020).