Self-Supervised Object Localization with Joint Graph Partition

Yukun Su^{1,3}, **Guosheng Lin**^{3†}, **Yiwen Cao**^{1,2}, **Qingyao Wu**^{1,4†}
[1]School of Software and Engineering, South China University of Technology
[2]Key Laboratory of Big Data and Intelligent Robot, Ministry of Education
[3] School of Computer Science and Engineering, Nanyang Technological University
[4] Pazhou Lab, Guangzhou, China
suyukun666@gmail.com, gslin@ntu.edu.sg, yiwen.cao@outlook.com, qyw@scut.edu.cn

Abstract

Object localization aims to generate a tight bounding box for the target object, which is a challenging problem that has been deeply studied in recent years. Since collecting bounding-box labels is time-consuming and laborious, many researchers focus on weakly supervised object localization (WSOL). As the recent appealing self-supervised learning technique shows its powerful function in visual tasks, in this paper, we take the early attempt to explore unsupervised object localization by self-supervision. Specifically, we adopt different geometric transformations to image and utilize their parameters as pseudo labels for self-supervised learning. Then, the class-agnostic activation map is used to highlight the target object potential regions. However, such attention maps merely focus on the most discriminative part of the objects, which will affect the quality of the predicted bounding box. Based on the motivation that the activation maps of different transformations of the same image should be equivariant, we further design a siamese network that encodes the paired images and propose a joint graph partition mechanism in an unsupervised manner to enhance the object co-occurrent regions. To validate the effectiveness of the proposed method, extensive experiments are conducted on CUB-200-2011, Stanford Cars and FGVC-Aircraft datasets. Experimental results show that our method outperforms state-of-the-art methods using the same level of supervision, even outperforms some weakly-supervised methods.

Introduction

Recently, deep convolution neural networks have achieved impressive results in many visual tasks such as recognition (Su et al. 2020) and segmentation (Chen et al. 2017), *etc*. This is due to the strong learning ability under supervision. Object localization aims to locate the object of interest within an image. However, collecting bounding-box labels is very time-consuming and labor-intensive, thereby some of these methods are often unavailable in practice.

To relax the demand for expensive annotations, some recent researches (Zhou et al. 2016; Choe and Shim 2019; Lin et al. 2020) focus on weakly-supervised object localization (WSOL), which only utilizes image-level labels. As the recent self-supervised learning techniques (He et al. 2020; Chen et al. 2020b,a; Gidaris, Singh, and Komodakis 2018) arising, image visual representations can be learned without labels in an unsupervised way. To this end, we make the early attempt to explore unsupervised object localization with self-supervised learning, which not only outperforms previous unsupervised works (Zhang et al. 2018a; Zhao et al. 2020), but also even outperforms some weakly-supervised methods (Zhou et al. 2016; Zhang et al. 2018b; Baek, Lee, and Shim 2020) (see in Figure 1).

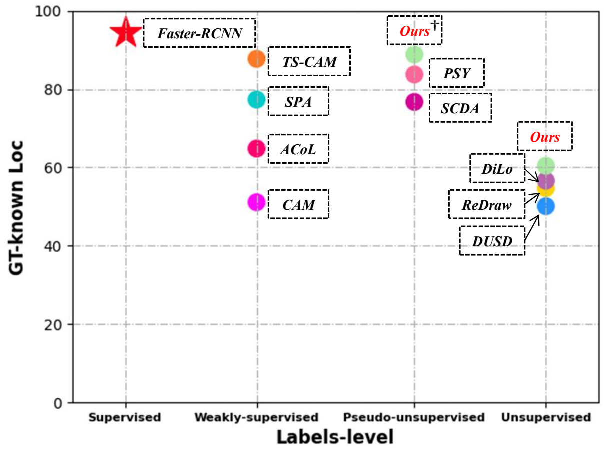
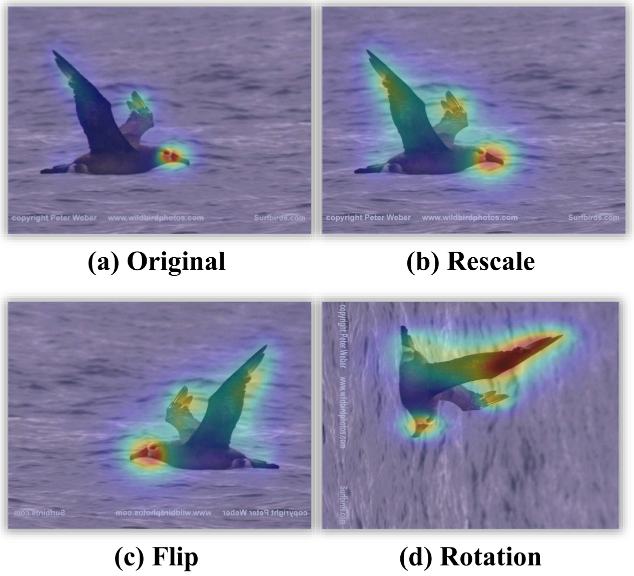


Figure 1: CUB-200-2011 (Wah et al. 2011) GT-Known *Loc* accuracy of different methods by different level supervisions. All methods adopt VGG16 as backbone for a fair comparison. '†' denotes the backbone is pre-trained on ImageNet (Russakovsky et al. 2015) using class-level labels.

In this work, we introduce a novel and simple framework for unsupervised object localization by self-supervision. Specifically, inspired by RotNet (Gidaris, Singh, and Komodakis 2018), the network can extract visual features by a classification task. The core intuition is that it is essentially impossible for a network to effectively perform the recognition task unless it has first learned to recognize and detect classes of objects as well as their semantic parts in images. But different from it, we consider predicting simple tasks may lead to degenerate learning (Jenni, Jin, and

†Corresponding authors.


(a) Original

(b) Rescale

(c) Flip

(d) Rotation

Figure 2: Inconsistent activation maps of different transformations (*i.e.*, original, rescale 2×, flipping and rotation 90°) of the same image.

Favaro 2020). Therefore, we adopt several different geometric transformations (*i.e.*, rotation, flipping, rescale and translation. Note that we can perform the regional dropout by erasing the parts of objects during transformation through affine padding.) to images and then use their parameters as artificial labels for self-supervised learning. In this way, the network model can learn to extract meaningful representations by predicting those transformations.

However, the activation maps generated from the aforementioned learning paradigm merely focus on the most discriminative part of the objects, which will affect the quality of the predicted bounding box. As shown in Figure 2, we observe that the activation maps of different geometric transformations of the same object distribute are in different parts (*i.e.*, head, wing), which is quite different from our human visual system. When we recognize the object, we can highlight the consistent part no matter it transforms. Motivated by this, we can utilize the divergence information for complementary learning to enhance the object activated regions. Specifically, we propose to encode the paired images with different transformations. Since the same object shares similar semantic information, therefore, we can propagate the learning features across the object. To this end, we introduce an unsupervised joint graph partition mechanism to mine the co-occurrent regions of the same object. By constructing and partitioning image feature graph obtained from paired images containing the same object, it yields the optimal masks for both the images that highlight the co-regions. Then we can use the partition results to enhance the network. Finally, the class-agnostic activation map is used to highlight the target objects for bounding-box prediction. Extensive experiments on three benchmark datasets give both quantitative and qualitative results, demonstrating the superiority of our approach. Our main contributions are the following:

- We take the early attempt to conduct unsupervised object localization by self-supervision, which to our best knowledge, has not been well explored.
- We introduce a network to encode paired images with different transformations and propose an unsupervised joint graph partition mechanism to enhance the co-occurrent regions of the target object.
- Extensive experimental results on three benchmark datasets show the effectiveness of our proposed method and it can outperform the state-of-the-art unsupervised methods by a large margin, even outperforms some weakly-supervised methods.

Related Work

Weakly-supervised Object Localization

Weakly-supervised Object Localization (WSOL) aims to learn the localization of objects with only image-level labels. The mainstream and representative methods for WSOL are based on CAM (Zhou et al. 2016), which produced localization maps by aggregating deep feature maps using a class-specific fully connected layer. However, such the CAM-based methods only discover small discriminative parts of objects. To tackle this drawback, EIL (Mai, Yang, and Luo 2020) and ADL (Choe and Shim 2019) proposed to mine the objects by integrating discriminative region mining and adversarial erasing in a single forward-backward propagation. ACoL (Zhang et al. 2018b) used multiple parallel classifiers that were trained adversarially. CutMix (Yun et al. 2019) and CDA (Su et al. 2021b) also explored the strategy and forced the network to focus on more relevant parts of objects. Besides, there are some works (Xue et al. 2019; Zhang, Wei, and Yang 2020a) focused on the intra and inter pixel-level correlations to help the network learn divergent activation maps. In recent work, TS-CAM (Gao et al. 2021) took the full advantage of the self-attention mechanism in visual transformer for long-range dependency extraction for object mining. SPA (Pan et al. 2021) proposed a two-stage approach to leverage the structure information incorporated in convolutional features for WSOL.

Self-supervised Learning

Self-supervised learning is an important technique in unsupervised tasks. By setting up different proxy-tasks, the network itself can learn meaningful image feature representations. (Larsson, Maire, and Shakhnarovich 2016) performed image colorization pretext to establish a mapping from objects to colors. In (Pathak et al. 2016), they learned objects features by predicting the missing parts of the images. In recent studies, some works (Noroozi and Favaro 2016; Wei et al. 2019a) tried to solve jigsaw problems to learn the information of different patches in the images. RotNet (Gidaris, Singh, and Komodakis 2018) proposed a simple rotation transformation and achieved remarkable results. Besides, video information is also widely used for training unsupervised models (Misra, Zitnick, and Hebert 2016; Pathak et al. 2017; Mahendran, Thewlis, and Vedaldi 2018). Recently, contrastive learning (Tian, Krishnan, and Isola 2020;

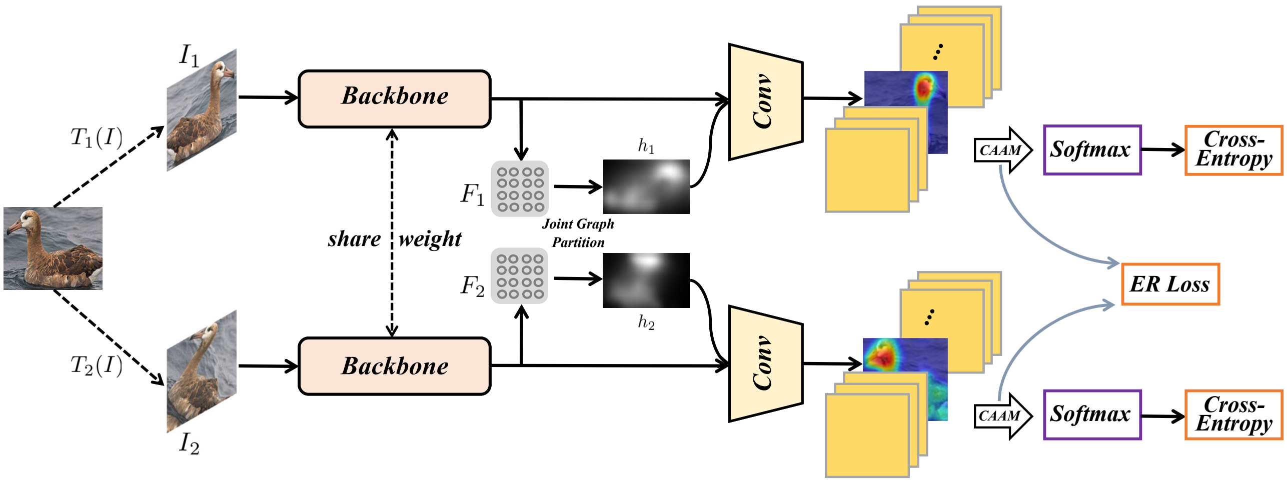


Figure 3: **The overview of our method**. Paired training images that contain the same object with different geometric transformations are first encoded into feature representations by a share-weight network. The joint graph partition module is responsible for dividing the two joint image features into two optimal sub-graphs and yields the masks to reinforce the co-occurrent object regions for enhancement followed by a standard convolutional layer. The final classification loss $\mathcal{L}_{cls}$ and equivariant regularization loss $\mathcal{L}_{ER}$ are used to update the gradient propagation to train the network in an end-to-end manner.

Su, Lin, and Wu 2021; Su et al. 2021a) by constructing pairs using different augmentations of image achieved great success. The core idea in contrastive learning is to strengthen the invariance of the network to various data augmentations. MoCo (He et al. 2020; Chen et al. 2020b) further improved the performance by using the memory bank and relaxing the big batch size for training. In this paper, we focus on self-supervised object localization that learns the potential image object feature representations with self-supervised strategy.

Unsupervised Object Discovery

Traditional unsupervised object learning methods are based on the similarity between image pixels such as superpixels (Wang et al. 2016) and Grabcut (Rother, Kolmogorov, and Blake 2004). In terms of deep learning, due to the need for a large number of ground-truth for training, there have been only a few unsupervised learning works in these years. (Cho et al. 2015) used off-the-shelf region proposals to form a set of candidate bounding boxes for objects and adopted probabilistic Hough transform to select the final prediction. MO (Zhang et al. 2019a), DDT (Wei et al. 2019b), SCDA (Wei et al. 2017a) and PSY (Baek, Lee, and Shim 2020) all utilized the models pre-trained on ImageNet (Russakovsky et al. 2015) that use class-level labels for post-process generation. Therefore, these are not strictly unsupervised object localization approaches. Another common method is to employ adversarial networks (Goodfellow et al. 2014) to find the objects. ReDraw (Chen, Artières, and Denoyer 2019) proposed to segment the objects by redrawing the masks of the targets based on an adversarial architecture. In this work, we compare our proposed method with the above mentioned unsupervised and weakly supervised methods to validate the effectiveness of our approach.

Methodology

We address unsupervised object localization in a self-supervised manner, and the core idea is to learn the potential object features that cover the entire region by predicting transformation task. In particular, we present a simple yet powerful framework to mine the object and propose an unsupervised joint graph partition strategy for object mining. In the following sections, we first give an overview of the entire architecture, and then introduce the joint graph partition method and objective functions in detail.

Overall Architecture

The entire framework of our proposed method is shown in Figure 3. Formally, given an unlabeled image I, we adopt two different random geometric transformations $\mathcal{T}_1$, $\mathcal{T}_2$ to it and form two new images I_1 and I_2. Note that the new images can also be the original image without transformations. We first adopt the backbone network (*i.e.*, VGG16 (Simonyan and Zisserman 2014)) as the encoder to extract their features denoted as F_1 and F_2, by removing the fully-connected layers and softmax layer. Since these two transformed images contain the same objects but with different parts of activation as we show in Figure 2, we then design a joint graph partition mechanism with learnable parameters, which is able to model the relationship between similar and dissimilar pixels for co-occurrent object regions mining for enhancement. The partition module yields two optimal solution masks that highlight the co-objects. Afterward, we add the masks to the former features F_1 and F_2 and followed by a 1×1 convolutional layer. Finally, we apply an average pooling along the channel axis to form the class-agnostic activation map. The prediction loss $\mathcal{L}_{cls}$ and equivariant regularization loss $\mathcal{L}_{ER}$ are used to train the network in an end-to-end manner.

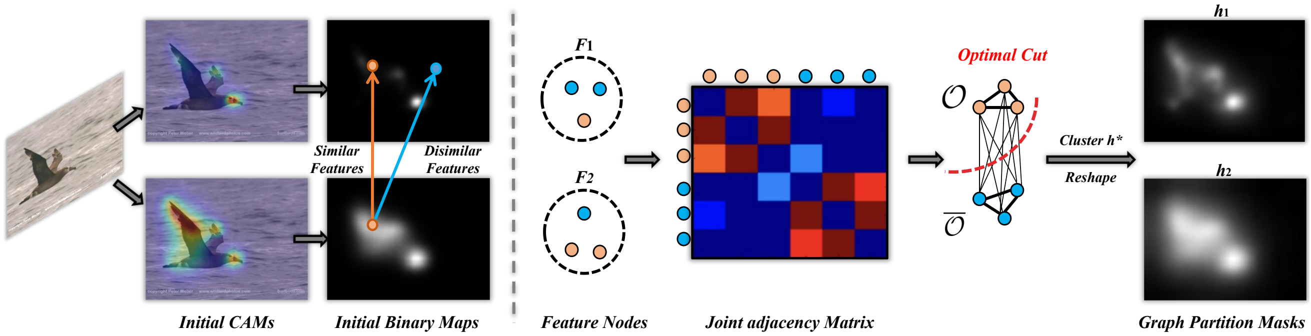


Figure 4: **Illustration of the joint graph partition module**. (Left) Visualization of the potential pixels in the initial activation maps. (Right) The paradigm of the unsupervised joint graph partition, which views the two feature maps as graph nodes and the edge represents the distance between each other. The similar object nodes are close while the dissimilar ones are distant. We aim to cut the graph and divide it into two optimal sub-graphs by using the subgraph indicator. The final optimal solution is then reshaped into two masks representing the co-occurrent regions.

Joint Graph Partition

As depicted in Figure 4 left, we transform the initial CAMs into grayscale maps, which show the potential foreground object pixels. We aim to use these two different maps to mine the semantic features in the co-occurrent part.

To this end, we propose a joint graph partition mechanism for better object mining as shown in Figure 4 right. Formally, the paired image feature maps $(F_1, F_2) \in \mathbb{R}^{w\times h\times c}$ are viewed as graph nodes, where w and h denote the spatial size of the feature map and c is the feature map dimension. We then construct a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents feature nodes in the graph, and $\mathcal{E}$ represents edges between nodes. In graph $\mathcal{G}$, we have $2wh$ nodes $N \in \mathbb{R}^{c\times 2wh}$ in total for paired images. We then use the paired images nodes to jointly construct a adjacency matrix A that represents the similarity of each node across inter and intra images. Specifically, we utilize Euclidean distance: $d_{i,j} = ||x_i - x_j||^2$ to measure the distance between the two arbitrary nodes in $\mathcal{G}$. Since we use normalized channel features for both F_1 and F_2, which satisfies $||x_i||_2^2 = 1$. By removing the constant, the adjacency matrix can be approximated to $A \in \mathbb{R}^{2wh\times 2wh} = N^TN$. As for the degree matrix D, since we consider both inter and intra information between paired images, nodes are fully connected between each other, $D = \text{diag}(\sum A_{i,j}, ..., \sum A_{i,j}) \in \mathbb{R}^{2wh\times 2wh}$. After converting the image features to a graph, we then cut the graph. Our goal is to make the different subgraphs as far apart as possible from each other and as similar internally as possible. In this way we can aggregate the co-occurrent category of prospects for subsequent enhancement. Inspired by (Ng, Jordan, and Weiss 2002), we will have the following function:

$$\mathcal{L}_m = \text{RatioCut}(\mathcal{V}_i, ..., \mathcal{V}_k) = \sum_i^k \frac{W(\mathcal{V}_i, \overline{\mathcal{V}}_i)}{|\mathcal{V}_i|}, \tag{1}$$

where $\{\mathcal{V}_1, \mathcal{V}_2, ..., \mathcal{V}_k\}$ represents a subset of $\mathcal{V}$. In our task, we aim to divide the foreground object $\mathcal{O}$ and background $\overline{\mathcal{O}}$, thus, $k = 2$. $W(\mathcal{O}, \overline{\mathcal{O}})$ represents the sum of edges of $\mathcal{O}$ and its complementary set, which equals to $\sum_{m\in\mathcal{O},n\in\overline{\mathcal{O}}} A_{m,n}$. $|\mathcal{O}|$ is the nodes cardinality in $\mathcal{O}$ subset.

In order to solve the optimal graph cut problem and avoid the occurrence of a single sample node as a subset, we then introduce the subgraph indicator $h = [h_1, ..., h_{2wh}]^T \in \mathbb{R}^{2\times w\times h}$ as follows:

$$h_i = \begin{cases} \frac{1}{\sqrt{|\mathcal{O}|}} & , \text{ if } \mathcal{V}_i \in \mathcal{O} \\ 0 & , \text{ if } \mathcal{V}_i \notin \mathcal{O} \end{cases} \tag{2}$$

Since D is a diagonal matrix, only the elements on the diagonal multiplied by the vector h_i have a value, where $D_{m,m} = \sum_{n=1} A_{m,n}$. And known from the property of the Laplace matrix, we then can put Eq 2 into Eq 1. The function $\mathcal{L}_m$ can be reformulated as:

$$\begin{aligned} \mathcal{L}_m &= \frac{1}{2}(\sum_{m\in\mathcal{O},n\in\overline{\mathcal{O}}} A_{m,n}\frac{1}{|\mathcal{O}|} + \sum_{m\in\overline{\mathcal{O}},n\in\mathcal{O}} A_{m,n}\frac{1}{|\mathcal{O}|}) \\ &= \frac{1}{2}(\sum_{m\in\mathcal{O},n\in\overline{\mathcal{O}}} A_{m,n}(\frac{1}{\sqrt{|\mathcal{O}|}} - 0)^2 \\ &+ \sum_{m\in\overline{\mathcal{O}},n\in\mathcal{O}} A_{m,n}(0 - \frac{1}{\sqrt{|\mathcal{O}|}}))^2 \\ &= \sum_{m=1}\sum_{n=1} h_m h_n D_{m,n} - \sum_{m=1}\sum_{n=1} h_m h_n A_{m,n} \\ &= h^T(D - A)h. \end{aligned} \tag{3}$$

Following (Zhang et al. 2020) that the continuous solution of the indicator vector is the principal component, we can finally get the optimal co-occurrent mask h^* by optimizing Eq 3. Afterward, we reshape it into $\{h_1, h_2\} \in \mathbb{R}^{w\times h\times 1}$ as the enhancement features and add them to the F_1 and F_2 respectively followed by a new standard convolutional layer for information updating. Note that except for tensor addition for features fusion, other alternatives like multiplication and concatenation will be discussed in our ablation studies.

Objective Function

Prediction Loss. For self-supervised learning, we provide the network $\mathcal{N}(\cdot)$ pretext task to update the gradient propagation and learn the object features. Specifically, we adopt different geometric transformations including the unchanged original images, random rotation (90°, 180°, 270°), random flipping, random scaling (1/4, 1/2, 2, 4) size of the original ones and translation. Then, we use the Cross-Entropy loss for transformation prediction as follows:

$$\mathcal{L}_{cls} = \text{Cross-Entroy}(\mathcal{N}(\mathcal{T}_i(I)), y_i), \qquad (4)$$

where y_i is the artificial label of each transformation $\mathcal{T}_i$.

Equivariant Regularization Loss. At the last layer of our network, we adopt an average pooling along the channel axis (*i.e.*, P_1 = AvgPool($\mathcal{N}(\mathcal{T}_1(I))[B,:,w,h]$), B is batch size) to yield the class-agnostic activation map (CAAM) followed by the Softmax function for classification. To further guarantee the consistency of input paired images for learning, we propose an equivariance regularization loss for regularizing the network prediction as follows:

$$\mathcal{L}_{ER} = ||P_1 - P_2||_1. \qquad (5)$$

Training. Finally, we train our network in an end-to-end manner, all the network parameters are jointly learned by minimizing the following multi-task loss function:

$$\mathcal{L}_{total} = \mathcal{L}_{cls} + \mathcal{L}_{ER} + \mathcal{L}_{m}. \qquad (6)$$

Experiments

Experimental Setup

Datasets. To evaluate the proposed approach, three datasets are adopted, including CUB-200-2011 (Wah et al. 2011), FGVC-Aircraft (Maji et al. 2013) and Stanford Cars (Krause et al. 2013). Among them, CUB-200-2011 is the largest dataset that contains 200 categories of birds with 5,994 training images and 5,794 testing images. We strictly follow the train-list and test-list of the datasets for training and evaluation and the bounding box annotations are solely used for evaluation.

Metrics. Following previous methods (Baek, Lee, and Shim 2020; Choe and Shim 2019), we use GT-Known localization (*GT-Known Loc*): fraction of images for which the predicted bounding box has more than 50% IoU with the ground-truth box. Since we target on unsupervised object localization, we do not report *Top-1*/ *Top-5* classification accuracy.

Implementation Details

In this work, we implement the proposed framework with PyTorch and train on 2080Ti-GPUs. For fair comparisons, VGG16 (Simonyan and Zisserman 2014) is used as the backbone network. The input image was resized to 256 × 256 and then was randomly cropped to 224 × 224 using zero padding if needed. We use stochastic gradient descent (SGD) optimizer with initial learning rate of 0.001, momentum of 0.9 and batch size of 32 for the model. The weight decay is set to 0.004. For strict **unsupervised** setting, we train our network without using any labels. For **pseudo-unsupervised** setting, we use the backbone pre-trained on ImageNet (Russakovsky et al. 2015) using class-level labels following previous methods (Zhang et al. 2019a; Wei et al. 2019b; Baek, Lee, and Shim 2020) for a fair comparison. Note that for both settings, we do not use any annotations from the three aforementioned datasets for training.

Baseline	$\mathcal{L}_m$	$\mathcal{L}_{ER}$	*GT-Known Loc*
✔			54.6
✔	✔		57.9
✔	✔	✔	**60.4**

Table 1: The ablation study for each proposed loss of our method.

Rotation	Flip	Rescale	Translation	*GT-Known Loc*
✔				57.9
✔	✔			58.1
✔		✔		58.6
✔			✔	58.2
✔	✔	✔		60.2
✔	✔		✔	60.0
✔		✔	✔	60.2
✔	✔	✔	✔	**60.4**

Table 2: Experiments of various transformations for self-supervised learning. Aggregating different affine transformations can bring significant improvement.

Ablation Studies

In this section, we explore the effectiveness of each component in our proposed method. Since CUB-200-2011 is a more challenging dataset, we conduct ablative analysis on it with the VGG16 backbone pretrained without using class-level labels if there is no special declaration.

Comparison with Baseline. Table 1 gives an ablation study of each loss in our approach. Note that in our method, **baseline** denotes using only prediction loss $\mathcal{L}_{cls}$ and using single image for training since it does not have $\mathcal{L}_m$ for graph partition. As can be seen, paired images mining boosts the network performance by **3.3%**, which reveals that joint graph partition is effective. Furthermore, when we introduce to use equivariant regularization loss $\mathcal{L}_{ER}$, we can further improve the localization accuracy achieving **60.4%**. This shows the effectiveness of the proposed losses in our self-supervised object localization learning, which can help the network to mine and regularize the potential object features. Besides, compared with baseline, we insert the joint graph partition module into the network structure. Since the module works at the top layer with a relatively small spatial size, thus the overhead is marginal compared with the baseline network. More detailed information about the computation complexity can be referred to supplementary material.

Method	Backbone	Venue	CUB-200-2011	Cars	Aircraft
Weakly-supervised Methods.					
CAM (Zhou et al. 2016)	VGG16	CVPR'16	51.09	-	-
ACoL (Zhang et al. 2018b)	VGG16	CVPR'18.	64.86	-	-
ADL (Choe and Shim 2019)	VGG16	CVPR'19	75.41	-	-
I^{2}C (Zhang, Wei, and Yang 2020b)	InceptionV3	ECCV'20	72.60	-	-
GC-Net (Lu et al. 2020)	GoogLeNet	ECCV'20	75.30	-	-
TS-CAM (Gao et al. 2021)	Transformer	ICCV'21	**87.70**	-	-
SPA (Pan et al. 2021)	VGG16	CVPR'21	77.29	-	-
Pseudo-unsupervised Methods.					
SCDA (Wei et al. 2017b)	VGG16	TIP'17	76.79	90.96	94.91
DDT (Wei et al. 2019b)	VGG16	PR'19	82.26	71.33	92.53
MO (Zhang et al. 2019b)	VGG16	-	80.45	92.51	94.94
PSY (Baek, Lee, and Shim 2020)	VGG16	AAAI'20	83.78	96.61	95.59
Ours	VGG16	-	**88.83**	97.73	96.72
Ours	InceptionV3	-	86.31	**98.62**	**97.94**
Unsupervised Methods.					
UODL (Cho et al. 2015)	-	CVPR'15	**69.37**	**93.05**	36.23
DUSD* (Zhang et al. 2018a)	ResNet101	CVPR'18	50.15	62.74	64.81
ReDraw* (Chen, Artières, and Denoyer 2019)	GAN	NIPS'19	54.73	42.75	37.89
DiLo* (Zhao et al. 2020)	VGG16	AAAI'21	56.68	62.37	64.59
Ours	VGG16	-	60.40	70.37	**74.62**

Table 3: Comparison between our method and the previous state-of-the-arts in terms of *GT-Known Loc* performance on CUB-200-2011, Stanford Cars and FGVC-Aircraft datasets. '*' indicates our reimplemented results using their publicly released code since they do not report the results.

Method	*GT-Known Loc*
Multiplication	57.8
Concatenation	59.2
Addition (Ours)	**60.4**

Table 4: Experiments of various features fusion strategies.

Method	*Part*	*More*
VGG16 (Simonyan and Zisserman 2014)	21.91	10.53
Ours	16.52	7.73
InceptionV3 (Szegedy et al. 2016)	23.09	5.52
Ours	18.21	5.48

Table 5: Localization error statistics.

The Effect of Different Transformations. As we mentioned in the introduction, too simple pretext tasks may degrade the learning features. As shown in Table 2, we explore different numbers of transformations for network learning. Only using rotation means the network has totally 4 classes to predict (*i.e.*, self image, 90°, 180° and 270° rotated images, respectively), which achieves 57.9% *GT-Known Loc*. When we utilize more geometric transformations, it makes the network harder to predict the class. As we finally use four different transformations, we can yield the best results to **60.4%**. This also shows that different geometric transformations can drive the network to activate objects in different areas, and the variety of transformations can help the network to mine more useful object regions by our proposed joint graph partition and regularization strategies.

The Effect of Different Features Fusion. Table 4 gives different features fusion strategies in our network. Among them, features addition yields the best performance. We conjecture that because the optimal masks have zero values, matrix multiplication turns some features to zero sharply, which may affect the features learning. And the concatenation operation can not well integrate the features globally.

Error Analysis. To further reveal the effect of our method, we categorize the localization errors into localization part error (*Part*) and localization more error (*More*). *Part* indicates that the predicted bounding box only covers parts of the object, and IoU is less than a certain threshold. On the contrary, *More* indicates that the predicted bounding box is larger than the ground truth bounding box by a large margin. Table 5 lists localization error statistics. Our method effectively reduces *Part*, and *More* errors using different backbones, which indicates that our localization maps are much accurate. More detailed definitions of each metric can be referred to supplementary material.

Comparisons with State-of-the-arts

We compare the proposed approach with the state-of-the-arts on the *GT-Known Loc* performance by using tight bounding boxes. Table 3 reports the results of our method

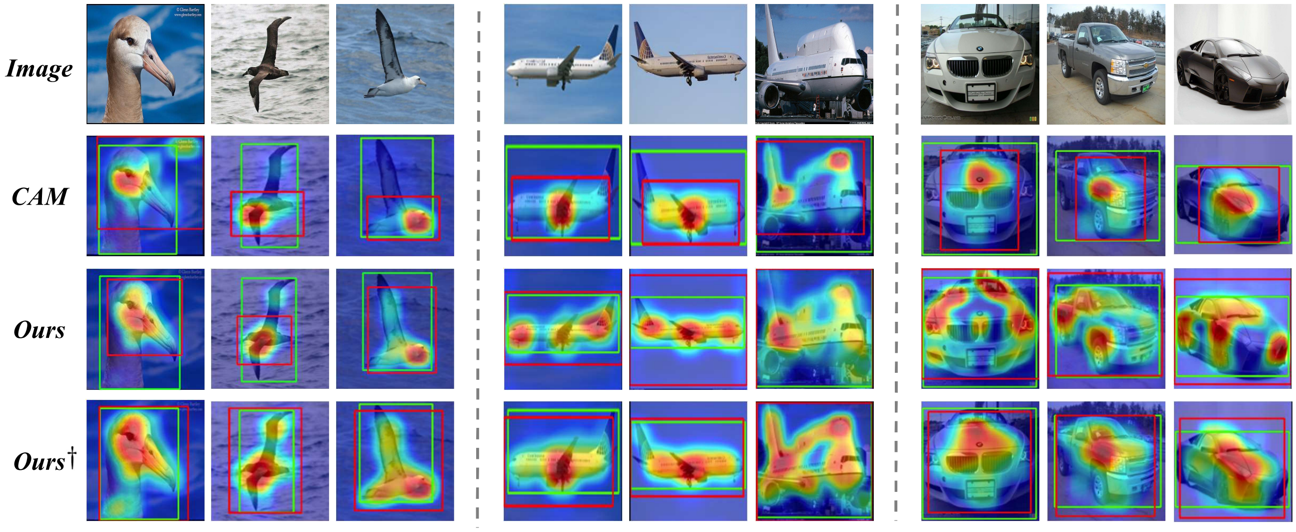


Figure 5: Visualization of the localization maps with CAM (Zhou et al. 2016) and our proposed method. The ground-truth bounding box is in green. The predicted bounding box is in red. Ours† denotes the backbone is pre-trained on ImageNet (Russakovsky et al. 2015) using class-level labels.

and several methods on the three benchmark datasets. To be specific, we first list some weakly-supervised object localization approaches at the top of Table 3. Among them TS-CAM (Gao et al. 2021) yields the best result achieving 87.70% accuracy on CUB-200-2011 dataset, outperforming the latest SPA (Pan et al. 2021) by 10.3% and achieves more than 30% gains over the CAM (Zhou et al. 2016) in terms of *GT-Known Loc*. It employs a visual transformer (Dosovitskiy et al. 2020) backbone network structure, which can help the network to learn the features globally.

Secondly, we compare our method with several methods under pseudo-unsupervised setting. As can be seen in the middle of Table 3, using the same VGG16 backbone, our method outperforms other SOTA methods by a large margin. On CUB-200-2011 dataset, we achieve **88.83%** *GT-Known Loc* performance gains compared with the second-best method PSY (Baek, Lee, and Shim 2020). On Stanford Cars and Aircraft datasets, we get remarkable performance achieving **97.73%** and **96.72%**, respectively. When we adopt a stronger InceptionV3 (Szegedy et al. 2016) backbone, we can further improve the *GT-Known Loc* performance on Stanford Cars and Aircraft datasets achieving **98.62%** and **97.94%**. It is worth mentioning that our proposed method even outperforms the best weakly-supervised methods (Ours: **88.83%** *vs.* TS-CAM: **87.70%**) on CUB-200-2011 dataset. Although the backbone is pretrained using class-level labels, compared with the WOSL methods, we train the network by self-supervision without using annotations on the training datasets and achieve significant performance. This validates the effectiveness of our proposed method and further ease the burden of using class labels for training the networks.

Finally, we show the unsupervised learning results at the bottom of Table 3. Since there are few researches on unsupervised object localization, we adopt similar unsupervised saliency detection method DUSD (Zhang et al. 2018a) and unsupervised segmentation method ReDraw (Chen, Artières, and Denoyer 2019) in our task. We also compare our method with a recently published approach DiLo (Zhao et al. 2020), which utilizes distilling localization for self-supervised representation learning. As can be seen, the traditional method UODL (Cho et al. 2015) yields the best results on both CUB-200-2011 and Stanford Cars datasets by using off-the-shelf region proposals. As for deep learning methods, our approach achieves the best results **74.62%** on Aircraft dataset. Besides, we achieve the second best performance on CUB-200-2011 approaching **60.4%** and Stanford Cars approaching **70.37%** in terms of *GT-Known Loc* without using other auxiliary techniques.

Visualization comparisons of the proposed approach are shown in Figure 5. Compared with the WSOL method CAM (Zhou et al. 2016), our method can mine more complete object regions than only focuses on the most discriminative ones. Bounding boxes produced by our method not only localize object regions accurately but also are more compact, which verifies its superiority.

Conclusion

In this paper, we take the early attempt to explore unsupervised object localization by self-supervision. By providing different geometric transformation pretext tasks and introducing a novel joint graph partition module, we encode paired images with the same object and mine the co-occurrent regions for features learning. To validate the effectiveness of our approach, extensive experiments are conducted on three benchmark datasets, and the results show that our method outperforms state-of-the-art methods.

Acknowledgments

This work was supported by National Natural Science Foundation of China (NSFC) 61876208, Key-Area Research and Development Program of Guangdong Province 2018B010108002, and the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG-RP-2018-003), and the MOE Tier-1 research grants: RG28/18 (S), RG22/19 (S) and RG95/20.

References

Baek, K.; Lee, M.; and Shim, H. 2020. Psynet: Self-supervised approach to object localization using point symmetric transformation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 10451–10459. 1, 3, 5, 6, 7

Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4): 834–848. 1

Chen, M.; Artières, T.; and Denoyer, L. 2019. Unsupervised object segmentation by redrawing. In *Advances in Neural Information Processing Systems*, 12726–12737. 3, 6, 7

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020a. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR. 1

Chen, X.; Fan, H.; Girshick, R.; and He, K. 2020b. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*. 1, 3

Cho, M.; Kwak, S.; Schmid, C.; and Ponce, J. 2015. Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1201–1210. 3, 6, 7

Choe, J.; and Shim, H. 2019. Attention-based dropout layer for weakly supervised object localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2219–2228. 1, 2, 5, 6

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*. 7

Gao, W.; Wan, F.; Pan, X.; Peng, Z.; Tian, Q.; Han, Z.; Zhou, B.; and Ye, Q. 2021. TS-CAM: Token Semantic Coupled Attention Map for Weakly Supervised Object Localization. *arXiv preprint arXiv:2103.14862*. 2, 6, 7

Gidaris, S.; Singh, P.; and Komodakis, N. 2018. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*. 1, 2

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial networks. *Advances in neural information processing systems*, 3(06). 3

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9729–9738. 1, 3

Jenni, S.; Jin, H.; and Favaro, P. 2020. Steering Self-Supervised Feature Learning Beyond Local Pixel Statistics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6408–6417. 1

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, 554–561. 5

Larsson, G.; Maire, M.; and Shakhnarovich, G. 2016. Learning representations for automatic colorization. In *European conference on computer vision*, 577–593. Springer. 2

Lin, C.; Wang, S.; Xu, D.; Lu, Y.; and Zhang, W. 2020. Object Instance Mining for Weakly Supervised Object Detection. In *AAAI*, 11482–11489. 1

Lu, W.; Jia, X.; Xie, W.; Shen, L.; Zhou, Y.; and Duan, J. 2020. Geometry constrained weakly supervised object localization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, 481–496. Springer. 6

Mahendran, A.; Thewlis, J.; and Vedaldi, A. 2018. Cross pixel optical-flow similarity for self-supervised learning. In *Asian Conference on Computer Vision*, 99–116. Springer. 2

Mai, J.; Yang, M.; and Luo, W. 2020. Erasing integrated learning: A simple yet effective approach for weakly supervised object localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8766–8775. 2

Maji, S.; Rahtu, E.; Kannala, J.; Blaschko, M.; and Vedaldi, A. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*. 5

Misra, I.; Zitnick, C. L.; and Hebert, M. 2016. Shuffle and learn: unsupervised learning using temporal order verification. In *European Conference on Computer Vision*, 527–544. Springer. 2

Ng, A. Y.; Jordan, M. I.; and Weiss, Y. 2002. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, 849–856. 4

Noroozi, M.; and Favaro, P. 2016. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, 69–84. Springer. 2

Pan, X.; Gao, Y.; Lin, Z.; Tang, F.; Dong, W.; Yuan, H.; Huang, F.; and Xu, C. 2021. Unveiling the Potential of Structure Preserving for Weakly Supervised Object Localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11642–11651. 2, 6, 7

Pathak, D.; Girshick, R.; Dollár, P.; Darrell, T.; and Hariharan, B. 2017. Learning features by watching objects move. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2701–2710. 2

Pathak, D.; Krahenbuhl, P.; Donahue, J.; Darrell, T.; and Efros, A. A. 2016. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2536–2544. 2

Rother, C.; Kolmogorov, V.; and Blake, A. 2004. " GrabCut" interactive foreground extraction using iterated graph cuts. *ACM transactions on graphics (TOG)*, 23(3): 309–314. 3

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3): 211–252. 1, 3, 5, 7

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. 3, 5, 6

Su, Y.; Lin, G.; Sun, R.; Hao, Y.; and Wu, Q. 2021a. Modeling the Uncertainty for Self-supervised 3D Skeleton Action Representation Learning. In *Proceedings of the 29th ACM International Conference on Multimedia*, 769–778. 3

Su, Y.; Lin, G.; and Wu, Q. 2021. Self-Supervised 3D Skeleton Action Representation Learning With Motion Consistency and Continuity. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 13328–13338. 3

Su, Y.; Lin, G.; Zhu, J.; and Wu, Q. 2020. Human Interaction Learning on 3D Skeleton Point Clouds for Video Violence Recognition. In *European Conference on Computer Vision*, 74–90. Springer. 1

Su, Y.; Sun, R.; Lin, G.; and Wu, Q. 2021b. Context Decoupling Augmentation for Weakly Supervised Semantic Segmentation. *arXiv preprint arXiv:2103.01795*. 2

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818–2826. 6, 7

Tian, Y.; Krishnan, D.; and Isola, P. 2020. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, 776–794. Springer. 2

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset. 1, 5

Wang, W.; Shen, J.; Shao, L.; and Porikli, F. 2016. Correspondence driven saliency transfer. *IEEE Transactions on Image Processing*, 25(11): 5025–5034. 3

Wei, C.; Xie, L.; Ren, X.; Xia, Y.; Su, C.; Liu, J.; Tian, Q.; and Yuille, A. L. 2019a. Iterative reorganization with weak spatial constraints: Solving arbitrary jigsaw puzzles for unsupervised representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1910–1919. 2

Wei, X.-S.; Luo, J.-H.; Wu, J.; and Zhou, Z.-H. 2017a. Selective convolutional descriptor aggregation for fine-grained image retrieval. *IEEE Transactions on Image Processing*, 26(6): 2868–2881. 3

Wei, X.-S.; Luo, J.-H.; Wu, J.; and Zhou, Z.-H. 2017b. Selective convolutional descriptor aggregation for fine-grained image retrieval. *IEEE Transactions on Image Processing*, 26(6): 2868–2881. 6

Wei, X.-S.; Zhang, C.-L.; Wu, J.; Shen, C.; and Zhou, Z.-H. 2019b. Unsupervised object discovery and co-localization by deep descriptor transformation. *Pattern Recognition*, 88: 113–126. 3, 5, 6

Xue, H.; Liu, C.; Wan, F.; Jiao, J.; Ji, X.; and Ye, Q. 2019. Danet: Divergent activation for weakly supervised object localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6589–6598. 2

Yun, S.; Han, D.; Oh, S. J.; Chun, S.; Choe, J.; and Yoo, Y. 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6023–6032. 2

Zhang, J.; Zhang, T.; Dai, Y.; Harandi, M.; and Hartley, R. 2018a. Deep unsupervised saliency detection: A multiple noisy labeling perspective. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9029–9038. 1, 6, 7

Zhang, K.; Chen, J.; Liu, B.; and Liu, Q. 2020. Deep object co-segmentation via spatial-semantic network modulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 12813–12820. 4

Zhang, R.; Huang, Y.; Pu, M.; Zhang, J.; Guan, Q.; Zou, Q.; and Ling, H. 2019a. Mining objects: Fully unsupervised object discovery and localization from a single image. 3, 5

Zhang, R.; Huang, Y.; Pu, M.; Zhang, J.; Guan, Q.; Zou, Q.; and Ling, H. 2019b. Mining Objects: Fully Unsupervised Object Discovery and Localization From a Single Image. *arXiv preprint arXiv:1902.09968*. 6

Zhang, X.; Wei, Y.; Feng, J.; Yang, Y.; and Huang, T. S. 2018b. Adversarial complementary learning for weakly supervised object localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1325–1334. 1, 2, 6

Zhang, X.; Wei, Y.; and Yang, Y. 2020a. Inter-image communication for weakly supervised localization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX 16*, 271–287. Springer. 2

Zhang, X.; Wei, Y.; and Yang, Y. 2020b. Inter-image communication for weakly supervised localization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX 16*, 271–287. Springer. 6

Zhao, N.; Wu, Z.; Lau, R. W.; and Lin, S. 2020. Distilling localization for self-supervised representation learning. *arXiv preprint arXiv:2004.06638*. 1, 6, 7

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2921–2929. 1, 2, 6, 7