Deep Object Co-Segmentation

Weihao Li*, Omid Hosseini Jafari*, Carsten Rother

Visual Learning Lab
Heidelberg University (HCI/IWR)
http://vislearn.de

Abstract. This work presents a deep object co-segmentation (DOCS) approach for segmenting common objects of the same class within a pair of images. This means that the method learns to ignore common, or uncommon, background *stuff* and focuses on *objects*. If multiple object classes are presented in the image pair, they are jointly extracted as foreground. To address this task, we propose a CNN-based Siamese encoder-decoder architecture. The encoder extracts high-level semantic features of the foreground objects, a mutual correlation layer detects the common objects, and finally, the decoder generates the output foreground masks for each image. To train our model, we compile a large object co-segmentation dataset consisting of image pairs from the PASCAL VOC dataset with common objects masks. We evaluate our approach on commonly used datasets for co-segmentation tasks and observe that our approach consistently outperforms competing methods, for both seen and unseen object classes.

Keywords: object co-segmentation, convolutional neural networks (CNN), end-to-end training

1 Introduction

Object co-segmentation is the task of detecting and segmenting the common objects from a set of images. This is used and applied in various computer vision applications and beyond, such as browsing in photo collections [1], 3D reconstruction [2], semantic segmentation [3], object-based image retrieval [4], video object tracking and segmentation [1], and interactive image segmentation [1].

There are different challenges for object co-segmentation with varying level of difficulty: (1) Rother *et al*. [1] first proposed the term of *co-segmentation* as the task of segmenting the common parts of an image pair simultaneously. They showed that segmenting two images jointly achieves better accuracy in contrast to segmenting them independently. They assume that the common objects are very similar, in terms of small appearance deviation, *i.e*. low intra-class variability. However, the background in both images can differ arbitrarily, see Fig. 1(a). (2) Another challenge is to segment common objects from the same class with low intra-class variation but similar background, see Fig. 1(b). (3) A more challenging task is to segment common objects from the same

* Equal contribution

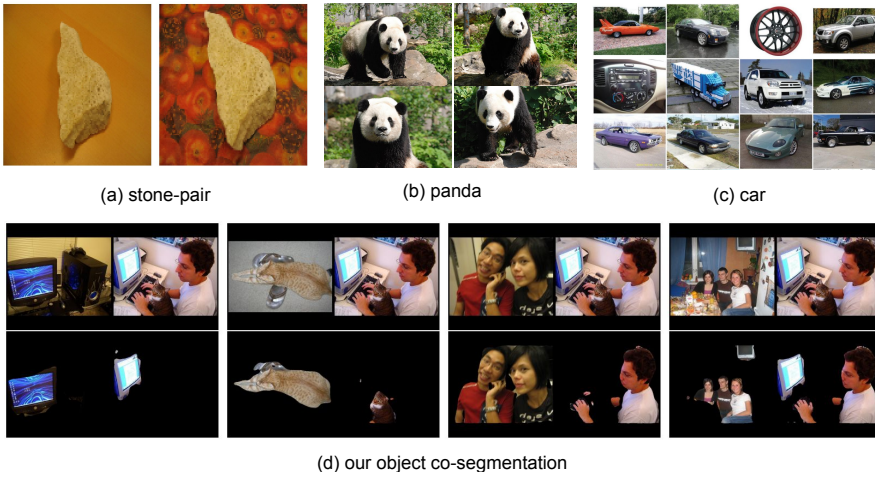

Fig. 1. (Top) Three different co-segmentation challenges: (a) segment common objects, in terms of small appearance deviation, with varying background (see [1]), (b) segment common objects from the same class with low intra class variation but similar background, (c) segment common objects from the same class with large variability in terms of scale, appearance, pose, viewpoint and (potentially) background. In this work we address the last case. (Bottom) A selection of image pairs from our PASCAL co-segmentation dataset and the corresponding results of our object co-segmentation approach.

class with large variability in terms of scale, appearance, pose, viewpoint and (potentially) background, see Fig. 1(c). In this work, we address the latter. Furthermore, most co-segmentation methods assume that every image to be co-segmented contains a single common object class. Here, we address the problem of co-segmentation without this assumption, *i.e*. multiple object classes can be presented within the images, see Fig. 1(d). Additionally, we are interested in co-segmenting objects with respect to *stuff*. The idea of object co-segmentation was introduced by Vicente *et al*. [4], to emphasize the resulting segmentation to be of an *object* such as a bird or a car, which excludes common, or uncommon, *stuff* classes like "sky" or "sea".

Segmenting objects in an image is one of the fundamental tasks in computer vision. While image segmentation has received great attention during the recent rise of deep learning methods [5,6,7,8,9], the related task of object co-segmentation remains largely unexplored by newly developed deep learning techniques. Most of the recently proposed object co-segmentation methods still rely on models with little or no feature learning. This includes methods utilizing super-pixels, or proposal segments [4,10] to extract a set of object candidates, or methods which use a complex CRF model [11,9] with hand-crafted features [9] to find the segments with the highest similarity.

In this paper, we propose a simple yet powerful method for segmenting objects of a common semantic class from a pair of images using a simple convolutional encoder-decoder neural network. Our method uses a pair of Siamese encoder networks to extract

semantic features for each image. The mutual correlation layer at the network's bottleneck computes localized correlations between the semantic features of the two images to predict the mask of common objects. Finally, the Siamese decoder networks combine the semantic features from each image with the correlation matrices and produce detailed segmentation masks through a series of deconvolutional layers. Our approach is trainable in an end-to-end manner and does not require any, potentially slow, CRF optimization procedure at evaluation time. We perform an extensive evaluation of our deep object co-segmentation and show that our model can achieve state-of-the-art performance on all common co-segmentation datasets. Additionally, we introduce a new PASCAL VOC co-segmentation dataset for training and testing our model, where we also achieve very good results.

In summary, our main contributions are:

- We propose a simple yet effective convolutional neural network architecture for object co-segmentation that can be trained end-to-end. To the best of our knowledge, this is the first *pure* fully convolutional neural networks framework for object co-segmentation, which does not depend on any hand-crafted features.
- We achieve state-of-the-art results on all common co-segmentation datasets and introduce a new challenging PASCAL object co-segmentation dataset for training and testing object co-segmentation models.

In the next section, we review related work on previous object co-segmentation approaches and recent approaches to image segmentation tasks using CNNs. Section 3 explains our deep object co-segmentation (DOCS) architecture design, loss functions, and extension for group object co-segmentation task. In Section 4.1 and 4.2 we describe our training details and training datasets. In Section 4.3 we present our experimental results on the MSRC, Internet, and iCoseg co-segmentation datasets.

2 Related Work

We start by discussing object co-segmentation by roughly categorizing them into three branches: co-segmentation without explicit learning, co-segmentation with learning and interactive co-segmentation. After that, we briefly discuss various image segmentation tasks and corresponding approaches based on convolutional neural networks and their relationships with to co-segmentation.

Co-Segmentation without Explicit Learning. Rother *et al*. [1] proposed the problem of image co-segmentation for image pairs. They minimize an energy function that combines an MRF smoothness prior term with a histogram matching term. This forces the histogram statistic of common foreground regions to be similar. In a follow-up work, Mukherjee *et al*. [12] replace the l_1 norm in the cost function by an l_2 norm. In Hochbaum and Singh [13] they used a reward model, in contrast to the penalty strategy of [1]. In [14], Vicente *et al*. studied various models and found that a simple model, based on Boykov-Jolly [15] to work best. Joulin *et al*. [16] formulated the co-segmentation problem in terms of a discriminative clustering task. Rubio *et al*. [17] proposed to match regions, which results from an over-segmentation algorithm, to establish correspondences between the common objects. Rubinstein *et al*. [18] combined

a visual saliency and dense correspondences, using SIFT flow, to capture the sparsity and visual variability of the common object in a group of images. Fu *et al*. [19] formulated object co-segmentation for RGB-D input images as a fully-connected graph structure, together with mutex constraints. In contrast to these works our method is a pure learning based approach.

Co-Segmentation with Learning. In [4], Vicente *et al*. generated a pool of object-like proposal-segmentations using constrained parametric min-cut [20]. Then they trained a random forest classifier to score the similarity of a pair of segmentation proposals. Yuan *et al*. [21] introduced a deep dense conditional random field framework for object co-segmentation by inferring co-occurrence maps. These co-occurrence maps measure the objectness scores, as well as, similarity evidence for object proposals, which are generated using selective search [22]. Similar to the constrained parametric min-cut, selective search also uses hand-crafted SIFT and HOG features to generate object proposals. Therefore, the whole model of [21] cannot be trained end-to-end. In addition, Yuan *et al*. assume that there is a single common object in a given image set, which limits its application in real-world scenarios. Recently, Quan *et al*. [9] proposed a manifold ranking algorithm for object co-segmentation by combining low-level appearance features and high-level semantic features. However, their semantic features are pre-trained on the ImageNet dataset. In contrast, our method is based on a pure CNN architecture, which is free of any hand-crafted features and object proposals and does not depend on any assumption about the existence of common objects.

Interactive Co-Segmentation. Batra *et al*. [23] firstly presented an algorithm for interactive co-segmentation of a foreground object from a group of related images. They use users' scribbles to indicate the foreground. Collins *et al*. [24] used a random walker model to add consistency constraints between foreground regions within the interactive co-segmentation framework. However, their co-segmentation results are sensitive to the size and positions of users' scribbles. Dong *et al*. [25] proposed an interactive co-segmentation method which uses global and local energy optimization, whereby the energy function is based on scribbles, inter-image consistency, and a standard local smoothness prior. In contrast, our work is not a user-interactive co-segmentation approach.

Convolutional Neural Networks for Image Segmentation. Convolutional neural networks have become the most popular method in many fields of computer vision, such as image recognition [26,27,28], object detection [29,30], optical flow [31]. In the last few years, convolutional neural networks (CNN) have achieved great success for the task of image segmentation, such as semantic segmentation [5,32,33,34,8,35], interactive segmentation [8,36], and salient object segmentation [37,38,39].

Semantic segmentation aims at assigning semantic labels to each pixel in an image. Fully convolutional networks (FCN) [5] became one of the first popular architectures for semantic segmentation. Nor *et al*. [32] proposed a deep deconvolutional network to learn the upsampling of low-resolution features. Both U-Net [6] and SegNet [40] proposed an encoder-decoder architecture, in which the decoder network consists of a hierarchy of decoders, each corresponding to an encoder. Yu *et al.* [33] and Chen *et al.* [41] proposed dilated convolutions to aggregate multi-scale contextual information, by

considering larger receptive fields. RefineNet [34] uses a multi-path refinement network with long-range residual connections to exploit multilevel features for high-resolution predictions. Zhang et al. [35] proposed a pyramid pooling module to exploit global context information.

Interactive segmentation enables users to select objects of interest by user interactions, such as a bounding box constraint. Xu *et al*. [8] proposed to solve the interactive segmentation problem using fully convolutional networks [5]. They transformed the user's input into Euclidean distance maps and then concatenated these maps as extra channels to train the fully convolutional network in an end-to-end fashion. Xu *et al*. [36] proposed a rectangle user input as a soft constraint, by transforming it into an Euclidean distance map. Then a convolutional encoder-decoder network is trained end-to-end by concatenating input images with these distance maps.

Salient object segmentation aims at detecting and segmenting the salient objects in a given image. Recently, deep learning architectures have become popular for salient object segmentation [37,38,39]. Li and Yu [37] addressed salient object segmentation using a deep network which consists of a pixel-level multi-scale fully convolutional network and a segment scale spatial pooling stream. Wang *et al*. [38] proposed recurrent fully convolutional networks to incorporate saliency prior knowledge for improved inference, utilizing a pre-training strategy based on semantic segmentation data. Jain *et al*. [39] proposed to train a deep fully convolutional neural network to produce pixel-level masks of all object-like regions given a single input image.

Although convolutional neural networks play a central role in image segmentation tasks, there has been no prior work with a pure CNN architecture for object co-segmentation. To the best of our knowledge, our deep convolutional neural network architecture is the first of its kind for object co-segmentation.

3 Method

In this section we introduce a new convolutional neural network architecture for segmenting the common objects from two input images. The architecture is end-to-end trainable for the object co-segmentation task. Fig. 2 illustrates the overall structure of our architecture. Our network consists of three main parts: (1) Given two input images I_A and I_B, we use a siamese encoder, which consists of convolutional layers, to extract high-level semantic feature maps f_A and f_B. (2) Then, we propose a mutual correlation layer to obtain correspondence maps c_{AB} and c_{BA} by matching feature maps f_A and f_B at pixel-level. (3) Finally, given the concatenations of the feature maps f_A and f_B and correspondence maps c_{AB} and c_{BA}, a siamese decoder, which consists of convolutional and deconvolution layers, is used to obtain and refine the common objects masks p_A and p_B.

In the following, we first describe each of the three parts of our architecture in detail. Then in Sec 3.4, the loss function is introduced. Finally in Sec 3.5, we explain how to extend our approach to handle co-segmentation of a group of images, *i.e*. going beyond two images.

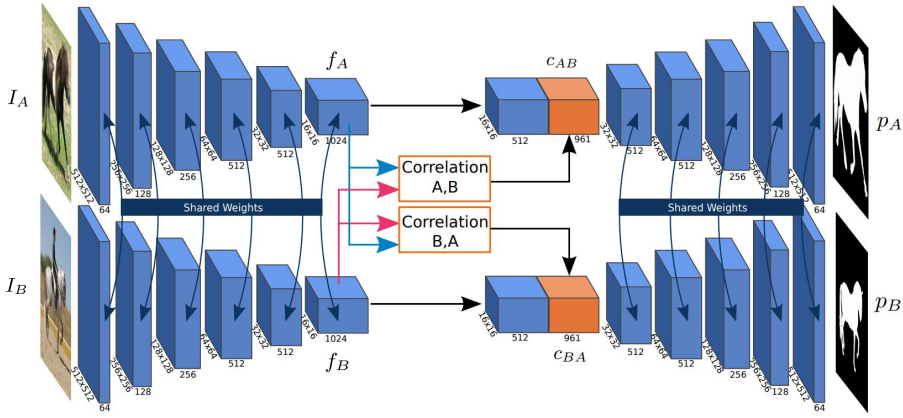

Fig. 2. Deep Object Co-Segmentation. We introduce a new convolutional neural network architecture for object co-segmentation for a given pair of input images. Our architecture includes three parts: (i) passing input images I_A and I_B through a siamese encoder consisting of convolutional layers to extract feature maps f_A and f_B, (ii) using a mutual correlation network to perform feature matching to obtain correspondence maps c_{AB} and c_{BA}, (iii) using a siamese decoder to get the common objects masks p_A and p_B, by combining the feature maps and correspondence maps.

3.1 Siamese Encoder

The first part of our architecture is a siamese encoder which consists of two identical feature extraction CNNs with shared parameters. We pass the input image pair I_A and I_B through the siamese encoder network pair, which is composed of convolutional layers to extract feature maps f_A and f_B. More specifically, our encoder is based on the VGG 16-layer network [27]. We keep the first 13 convolutional layers and remove the last two fully connected layers *fc6* and *fc7*, which are used for image classification. Instead, we replace *fc6* and *fc7* with two additional 3×3 convolutional layers *conv6-1* and *conv6-2* to produce feature maps which contain more spatial information. In total, our encoder network has 15 convolutional layers and 5 pooling layers in order to create a set of high-level semantic features f_A and f_B. The input to the siamese encoder is two 512×512 images and the output of encoder is two 1024-channel feature maps with a spatial size of 16×16.

3.2 Mutual Correlation

The second part of our architecture is a mutual correlation layer. The two feature maps f_A and f_B computed at the last layer of each encoder represent the high-level semantic content of the input images. This means that when the two images contain objects that belong to a common class, they should contain similar features at the locations of the shared objects. For this reason, we propose a mutual correlation layer to compute the

correlation between each pair of locations on the feature maps. The idea of utilizing the correlation layer is inspired by Flownet [31], in which the correlation layer is used to match feature points between frames for optical flow estimation.

In detail, the mutual correlation layer performs a pixel-wise comparison between two feature maps f_A and f_B. Given a point (i, j) and a point (m, n) inside a patch around (i, j), the correlation between feature vectors $f_A(i, j)$ and $f_B(m, n)$ is defined as

$$c_{AB}(i, j, k) = \langle f_A(i, j), f_B(m, n) \rangle \tag{1}$$

where $k = (n - j)D + (m - i)$ and $D \times D$ is patch size. Since, the common objects can locate at any place on the two input images, we set the patch size to $D = 2 * max(w, h) + 1$, where w and h are the width and height of the feature maps f_A and f_B. The output of the correlation layer is a feature map c_{AB} of size $w \times h \times D^2$. We use the same method to compute the correlation map c_{BA} between f_B and f_A.

3.3 Siamese Decoder

The siamese decoder is the third part of our architecture, which predicts the two foreground masks of the common objects. The input to the siamese decoder is achieved by concatenating the feature maps f_A and f_B and their corresponding correlation maps c_{AB} and c_{BA}, as shown in Fig. 2. To make the training feasible, it is necessary to use spatial pooling in the encoder networks. However, due to this, the feature map produced by the encoder is coarse. The decoder then enlarges the feature maps using deconvolutional layers in order to produce the final object segmentation result. Just like the siamese encoder, the decoder is also arranged in a siamese structure so that these two networks share their parameters. There are five blocks in our decoder, whereby each block has one deconvolutional layer and two convolutional layers. All the convolutional and deconvolutional layers in our siamese decoder are followed by a ReLU activation function. By applying a Softmax function, the decoder produces two probability maps p_A and p_B. Each probability map has two channels, background and foreground, with the same size as the input images.

3.4 Loss Function

We define our object co-segmentation as a binary image labeling problem and use the standard cross entropy loss function to train our network. The full loss score $\mathcal{L}_{A,B}$ is then estimated by

$$\mathcal{L}_{AB} = \mathcal{L}_A + \mathcal{L}_B \tag{2}$$

where the $\mathcal{L}_A$ and the $\mathcal{L}_B$ are cross-entropy loss functions for the image A and the image B, respectively.

3.5 Group Co-Segmentation

Although our architecture is trained for image pairs, we can extend our proposed co-segmentation method to handle a group of images. Given a set of N images $\mathcal{I} =$

$\{I_1, ..., I_N\}$, we use our DOCS network to predict the probability maps for all possible pairs in the image set, $\mathcal{P} = \left\{p_n^k : 1 \leq n, k \leq N, k \neq n\right\}$, where p_n^k is the predicted probability map for image I_n when it is paired with image I_k. Finally, we compute the final mask M_n for image I_n as

$$M_n(x, y) = \underset{k \neq n}{\text{median}}\{p_n^k(x, y)\} > 0.5. \tag{3}$$

We use the median to make our approach more robust to groups with outliers.

4 Experiments

Our proposed method is successful on the task of object co-segmentation. In this section, we first describe our training datasets and training procedure, then we show the results on several object co-segmentation datasets, including MSCR [42,14], Internet [18], and iCoseg [23].

4.1 Datasets

Training a deep neural network requires a lot of data. However, existing co-segmentation datasets are either are too small or have a limited number of object classes. The MSRC dataset [42] was first introduced for supervised semantic segmentation, then a subset was used for object co-segmentation [4]. This subset of MSRC only has 7 groups of images and each group has 10 images. The iCoseg dataset, introduced in [23], consists of several groups of images and is widely used to evaluate co-segmentation methods. However, each group contains images of the same object instance or very similar objects from the same class. The Internet dataset [18] contains thousands of images obtained from the Internet using image retrieval techniques. However, it only has three object classes: car, horse and airplane, where images of each class are mixed with other noise objects. In [43], Faktor and Irani use PASCAL VOC 2010 dataset for object co-segmentation. Due to the large variability in scale, appearance, pose, viewpoint and background of objects, using the PASCAL VOC dataset for object co-segmentation is challenging. Faktor and Irani [43] separate the images into 20 groups according to the object classes and assume that each group only has one object. However, this assumption is not common for natural images.

Inspired by [43], we generated a new object co-segmentation dataset using the PASCAL VOC 2012 dataset [44]. The original dataset consists of 20 foreground object classes and one background class. It contains 1464 (train) and 1449 (val) pixel-level labeled images for training and validation. From the original 1464 training images, we pick all pairs of images which have common objects and generate 169.085 image pairs as a new co-segmentation training set. We randomly split the original validation set into two groups, 725 images for test and 724 images for validation. For each group, we pick all pairs of images which have common objects and generate the co-segmentation test set with 40.303 image pairs and the co-segmentation validation set with 42.831 image pairs. Since our goal is to segment the common objects from the pair of images, we discard the 20 object class labels and instead label the common objects as foreground. Fig. 1 shows some examples of image pairs of our new co-segmentation dataset.

4.2 Training

We use the Caffe framework [45,31] to design and train our deep object co-segmentation network. To train our model, we only use our co-segmentation dataset, which is generated using the PASCAL VOC dataset. We did not use any images from the MSCR, Internet or iCoseg datasets to fine tune our model. The *conv1-conv5* layers of our siamese encoder (VGG-16 net [27]) are initialized with weights of a model pre-trained on the Imagenet image classification dataset [46], which has proven to be a good initialization for other vision tasks [5,32,30,37]. We optimize our network parameters with the Adam solver [47]. We use small mini-batches of 10 image pairs, a momentum of 0.9, a learning rate of $1e-5$, and a weight decay of 0.0005.

4.3 Results

We report the performance of our approach on MSCR [42,14], Internet [18] and iCoseg [23] datasets, as well as our own co-segmentation dataset. First, we introduce the metrics we used in this work. Then, we show our results visually and quantitatively on different datasets.

Metrics. For evaluating the co-segmentation performance, there are two common metrics used. The first one is *Precision*, which is the percentage of correctly segmented pixels of both the foreground and the background mask. The second one is *Jaccard*, which is the intersection over union of the co-segmentation result and the ground truth foreground segmentation.

Our PASCAL Co-Segmentation Dataset. Our new PASCAL co-segmentation dataset consists of 40303 test image pairs. It is very challenging to segment common objects in this dataset, since the objects vary in scale, appearance, pose and viewpoint. Each image pair has at least one common object. The same image can have different common objects when it is paired with different images, which makes the task of object co-segmentation more difficult.

We also tried to obtain the common objects of same classes using a deep semantic segmentation model, here FCN8s [5]. First, we train FCN8s with the original PASCAL VOC 2012 dataset. Then, we obtain the common objects from two images by predicting the semantic labels using FCN8s and keeping the segments with common classes as foreground. Our co-segmentation method (93.78% for *Precision* and 57.82% for *Jaccard*) outperforms FCN8s (93.20% for *Precision* and 55.21% for *Jaccard*), which uses the same VGG encoder, and trained with the same training images. The improvement is probably due to the fact that our DOCS architecture is specifically designed for the object co-segmentation task, which FCN8s is designed for the semantic labeling problem. Another potential reason is that generating image pairs is a form of data augmentation. We would like to exploit these ideas in the future work.

Fig. 3 shows qualitative results of our approach on the PASCAL co-segmentation dataset. In particular, in the first four rows, we can see that our method successfully extracts different foreground objects for the left image when paired with a different image to the right.

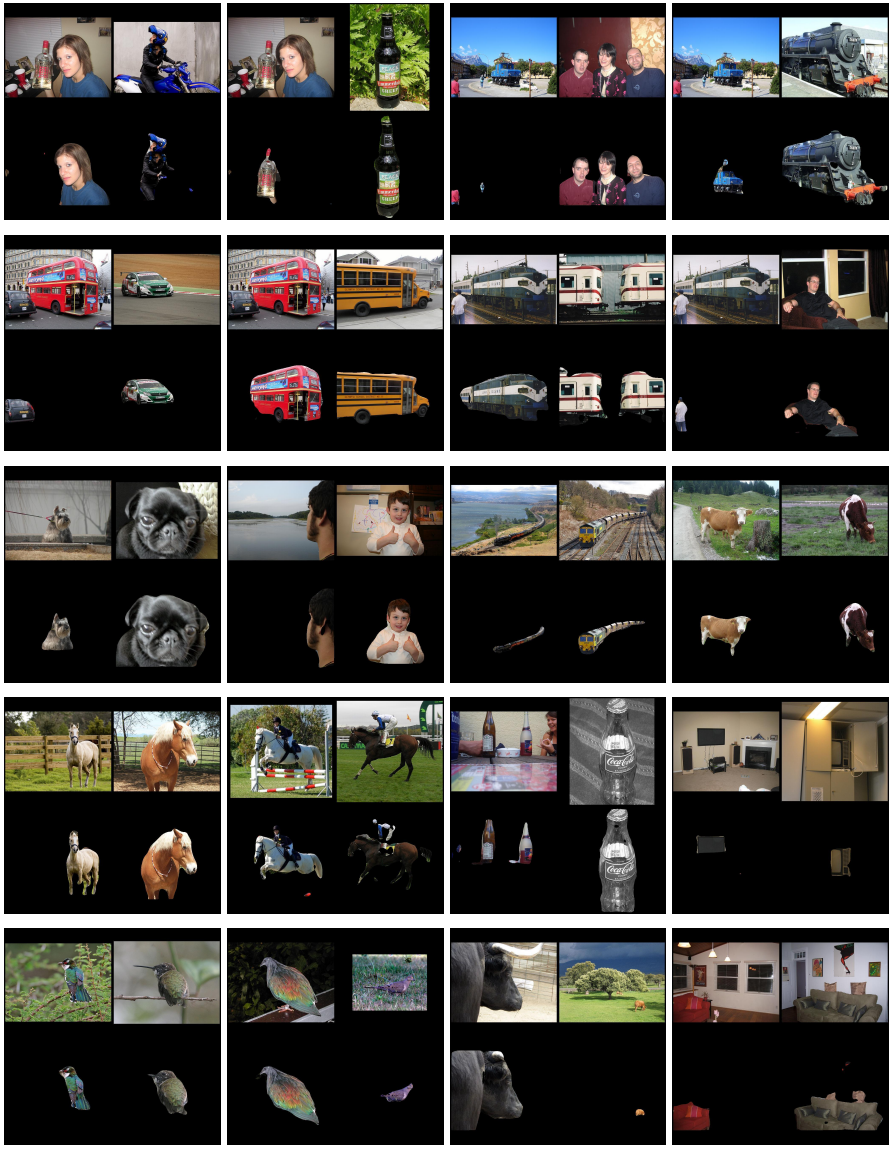

Fig. 3. Our qualitative results on PASCAL Co-segmentation dataset. Odd rows: the original image pairs. Even rows: the corresponding object co-segmentation results. In particular, in the first four rows we can see that our method successfully extracts different foreground objects for the left image when paired with a different image to the right.

Table 1. Quantitative results on the MSRC dataset (Seen classes). Quantitative comparison results of our DOCS approach with four state-of-the-art co-segmentation methods on the co-segmentation subset of the MSCR dataset.

MSCR	[4]	[18]	[48]	[43]	Ours
Precision	90.2	92.16	92.23	92.0	**94.43**
Jaccard	70.6	74.7	-	77.0	**79.89**

MSRC Dataset. We evaluate our method on the subset of the MSRC dataset, which has been widely used by previous methods to evaluate co-segmentation performance [4,18,43,48]. We select the same classes, which are cow, plane, car, sheep, bird, cat and dog, as reported by [4,18,43,48]. Each class group has 10 images and there is a single object in each image. The objects in each class have color, pose and scale differences. We use our group co-segmentation method to extract the foreground masks for each class group.

In Table. 1, we show the quantitative results of our method as well as four state-of-the-art methods [4,18,43,48]. Our *Precision* (94.43%) and *Jaccard* (79.89.1%) show a significant improvement compared to previous methods.

It is important to note that [4] and [48] are supervised methods, i.e. both use images of the MSRC dataset to train their models. We obtain the new state-of-the-art results on this dataset even without training or fine-tuning on any images from the MSRC dataset. Visual examples of object co-segmentation results on the subset of the MSRC dataset can be found in Fig. 4.

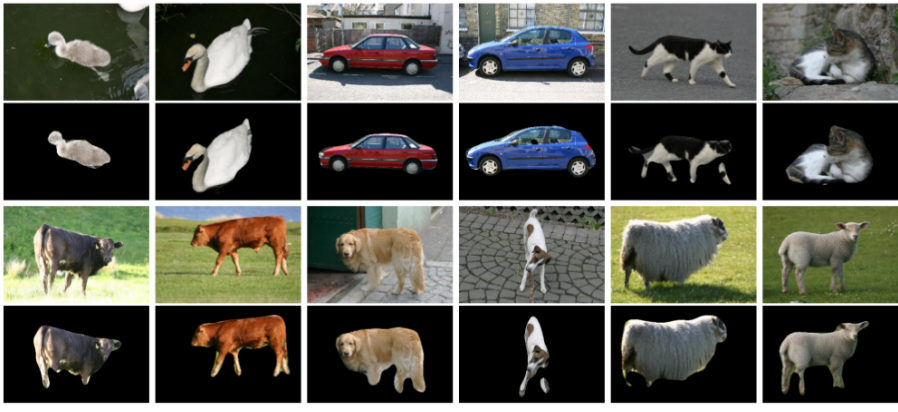

Fig. 4. Our qualitative results on the MSRC co-segmentation dataset (Seen classes). Odd rows: the original images. Even rows: the corresponding object co-segmentation results.

Internet Dataset. There are three image groups in the Internet dataset [18], which are airplane, horse and car. Each group consists of thousands of images. Most of the images have common objects with the same class and some "outlier" images do not contain the common objects. In our experiment, following [18,49,9,21], we utilize the subset of the Internet dataset, in which there are 100 images per class group. We use our group co-segmentation method to extract the foreground mask for each class.

We compare our method with five previous approaches [16,49,18,9,21] on this subset of the Internet dataset. Table 2 shows the quantitative results of each object class with respect to *Precision* and *Jaccard*. We outperform all previous methods [16,49,18,9,21] in terms of *Precision* measure and average of the *Jaccard* measure. Note that [21] is also a supervised co-segmentation method, in contrast to ours.

Fig. 5 shows some quantitative results of our method. It can be seen that our object co-segmentation method can detect and segment the common objects of each class accurately. Even for the "outlier" images in each group, our method can successfully recognize them. Here, the last column shows the "outlier" images.

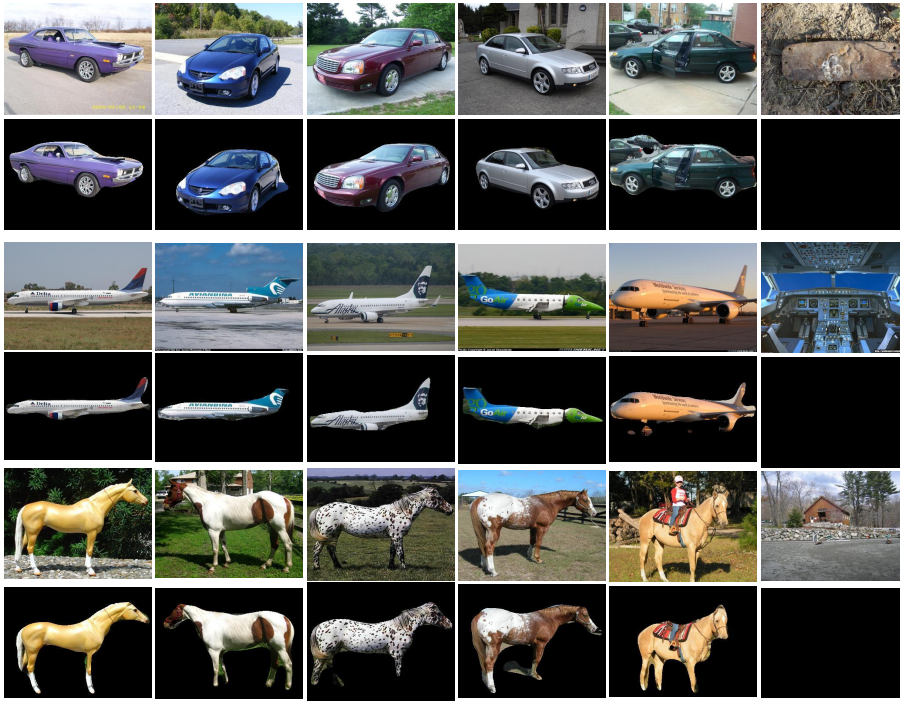

Fig. 5. Our qualitative results on the Internet dataset (Seen classes). Some results of our object co-segmentation method for the Internet dataset. We show the original images in the first row and the corresponding object co-segmentation results in the second row.

Table 2. Quantitative results on the Internet dataset (Seen classes). Quantitative comparison of our DOCS approach with several state-of-the-art co-segmentation methods on the co-segmentation subset of the Internet dataset. We outperform others in nearly all cases.

Internet		[16]	[18]	[49]	[9]	[21]	Ours
Car	P	58.7	85.3	87.6	88.5	90.4	**94.0**
	J	37.1	64.4	64.9	66.8	72.0	**82.7**
Horse	P	63.8	82.8	86.2	89.3	90.2	**91.4**
	J	30.1	51.6	33.4	58.1	**65.0**	64.6
Airplane	P	49.2	88.0	90.3	92.6	91.0	**94.6**
	J	15.3	55.8	40.3	56.3	**66.0**	63.5
Average	P	57.2	85.4	88.0	89.6	91.1	**93.3**
	J	27.5	57.3	46.2	60.4	67.7	**70.3**

iCoseg Dataset is widely used for evaluating co-segmentation methods. For a fair comparison with existing methods [50,18,43,51], which report the quantitative results of each group, we use the same setting of the dataset in our experiments. This includes 30 image groups and a total of 530 images. In contrast to the MSRC and Internet datasets, where object classes are part of the PASCAL VOC dataset, there are several object classes in the iCoseg dataset which do not appear in PASCAL VOC dataset. Therefore, it is possible to to use iCoseg dataset to evaluate the generalization of our method on *unseen object classes*. In order to test our method on unseen objects, we choose eight groups of images from the iCoseg dataset as our unseen object classes, which are bear2, brown_bear, cheetah, elephant, helicopter, hotballoon, panda1 and panda2. There are two reasons for this choice: firstly, these object classes are not included in the PASCAL VOC dataset, thus we ignore groups like baseball, gymnastic and women soccer, whose object classes already appear in PASCAL VOC. Secondly, in order to focus on *objects*, in contrast to "stuff", we ignore groups like pyramid, stonehenge and taj-mahal.

We compare our method with four state-of-the-art approaches [50,18,43,51] on unseen objects of the iCoseg dataset. Table 3 shows the comparison results of each unseen object groups in terms of *Jaccard*. The results show that for 6 out of 8 object groups our method performs best, and it is also superior on average. Note that the results of [50,18,43,51] are taken from Table X in [51].

Fig. 6 shows some quantitative results of our method. It can be seen that our object co-segmentation method can detect and segment the common objects of these unseen classes accurately.

5 Conclusion

In this work, we presented a new and efficient CNN-based method for solving the problem of object class co-segmentation, which consists of jointly detecting and segmenting objects belonging to a common semantic class from a pair of images. Based on a simple encoder-decoder architecture, combined with the mutual correlation layer for matching semantic features, we achieve state-of-the-art performance on various datasets, and

Table 3. Quantitative results on the iCoseg dataset (Unseen classes). Quantitative comparison of our DOCS approach with four state-of-the-art co-segmentation methods on some *object* classes of the iCoseg dataset, in terms of Jaccard. For this dataset, these object classes were not known during training of our method (i.e. "unseen").

iCoseg	[18]	[50]	[43]	[51]	Ours
bear2	65.3	70.1	72.0	67.5	**88.3**
brownbear	73.6	66.2	**92.0**	72.5	**92.0**
cheetah	69.7	75.4	67.0	**78.0**	68.8
elephant	68.8	73.5	67.0	79.9	**84.6**
helicopter	80.3	76.6	**82.0**	80.0	79.0
hotballoon	65.7	76.3	88.0	80.2	**91.7**
panda1	75.9	80.6	70.0	72.2	**82.6**
panda2	62.5	71.8	55.0	61.4	**86.7**
average	70.2	73.8	78.2	74.0	**84.2**

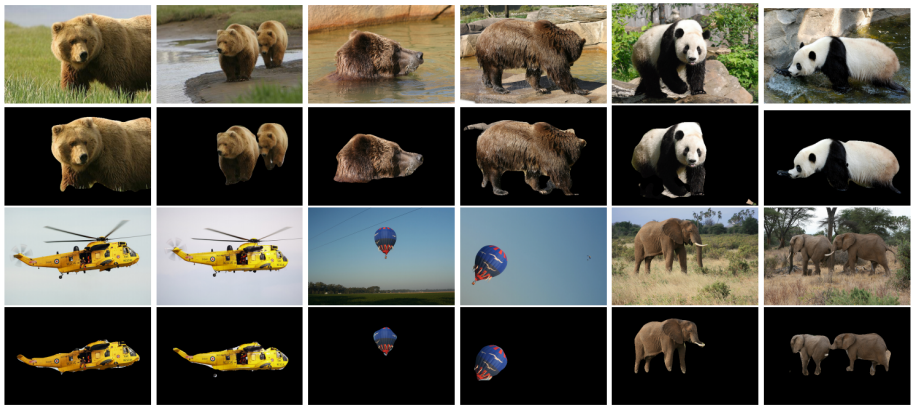

Fig. 6. Our qualitative results on iCoseg dataset (Unseen classes). Some results of our object co-segmentation method, with original image pairs in the first row and the corresponding object co-segmentation results in the second row. For this dataset, the object classes were not known during training of our method (i.e. "unseen").

demonstrate good generalization performance on segmenting objects of new semantic classes, unseen during training. To train our model, we compile a large object co-segmentation dataset consisting of image pairs from PASCAL VOC dataset with shared objects masks.

References

1. Rother, C., Minka, T., Blake, A., Kolmogorov, V.: Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In: CVPR. (2006)
2. Kowdle, A., Batra, D., Chen, W.C., Chen, T.: imodel: Interactive co-segmentation for object of interest 3d modeling. In: ECCV workshop. (2010)
3. Shen, T., Lin, G., Liu, L., Shen, C., Reid, I.: Weakly supervised semantic segmentation based on co-segmentation. In: BMVC. (2017)
4. Vicente, S., Rother, C., Kolmogorov, V.: Object cosegmentation. In: CVPR. (2011)
5. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR. (2015)
6. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI. (2015)
7. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., Torr, P.H.S.: Conditional random fields as recurrent neural networks. In: ICCV. (2015)
8. Xu, N., Price, B., Cohen, S., Yang, J., Huang, T.S.: Deep interactive object selection. In: CVPR. (2016)
9. Quan, R., Han, J., Zhang, D., Nie, F.: Object co-segmentation via graph optimized-flexible manifold ranking. In: CVPR. (2016)
10. Taniai, T., Sinha, S.N., Sato, Y.: Joint recovery of dense correspondence and cosegmentation in two images. In: CVPR. (2016)
11. Lee, C., Jang, W.D., Sim, J.Y., Kim, C.S.: Multiple random walkers and their application to image cosegmentation. In: CVPR. (2015)
12. Mukherjee, L., Singh, V., Dyer, C.R.: Half-integrality based algorithms for cosegmentation of images. In: CVPR. (2009)
13. Hochbaum, D.S., Singh, V.: An efficient algorithm for co-segmentation. In: ICCV. (2009)
14. Vicente, S., Kolmogorov, V., Rother, C.: Cosegmentation revisited: Models and optimization. In: ECCV. (2010)
15. Boykov, Y.Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In: ICCV. (2001)
16. Joulin, A., Bach, F., Ponce, J.: Discriminative clustering for image co-segmentation. In: CVPR. (2010)
17. Rubio, J.C., Serrat, J., López, A., Paragios, N.: Unsupervised co-segmentation through region matching. In: CVPR. (2012)
18. Rubinstein, M., Joulin, A., Kopf, J., Liu, C.: Unsupervised joint object discovery and segmentation in internet images. In: CVPR. (2013)
19. Fu, H., Xu, D., Lin, S., Liu, J.: Object-based rgbd image co-segmentation with mutex constraint. In: CVPR. (2015)
20. Carreira, J., Sminchisescu, C.: Constrained parametric min-cuts for automatic object segmentation. In: CVPR. (2010)
21. Yuan, Z., Lu, T., Wu, Y.: Deep-dense conditional random fields for object co-segmentation. In: IJCAI. (2017)
22. Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W.: Selective search for object recognition. IJCV (2013)
23. Batra, D., Kowdle, A., Parikh, D., Luo, J., Chen, T.: icoseg: Interactive co-segmentation with intelligent scribble guidance. In: CVPR. (2010)
24. Collins, M.D., Xu, J., Grady, L., Singh, V.: Random walks based multi-image segmentation: Quasiconvexity results and gpu-based solutions. In: CVPR. (2012)
25. Dong, X., Shen, J., Shao, L., Yang, M.H.: Interactive cosegmentation using global and local energy optimization. IEEE Transactions on Image Processing **24**(11) (2015) 3966–3977

26. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS. (2012)
27. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR. (2015)
28. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. (2016)
29. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR. (2014)
30. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NIPS. (2015)
31. Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., van der Smagt, P., Cremers, D., Brox, T.: Flownet: Learning optical flow with convolutional networks. In: ICCV. (2015)
32. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: ICCV. (2015)
33. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: ICLR. (2016)
34. Lin, G., Milan, A., Shen, C., Reid, I.: Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In: CVPR. (2017)
35. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR. (2017)
36. Xu, N., Price, B., Cohen, S., Yang, J., Huang, T.: Deep grabcut for object selection. In: BMVC. (2017)
37. Li, G., Yu, Y.: Deep contrast learning for salient object detection. In: CVPR. (2016)
38. Wang, L., Wang, L., Lu, H., Zhang, P., Ruan, X.: Saliency detection with recurrent fully convolutional networks. In: ECCV. (2016)
39. Jain, S.D., Xiong, B., Grauman, K.: Pixel objectness. arXiv preprint arXiv:1701.05349 (2017)
40. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for scene segmentation. TPAMI (2017)
41. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected crfs. In: ICLR. (2015)
42. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: ECCV. (2006)
43. Faktor, A., Irani, M.: Co-segmentation by composition. In: ICCV. (2013)
44. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html
45. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: ACM Multimedia. (2014)
46. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. (2009)
47. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. In: ICLR. (2015.)
48. Wang, F., Huang, Q., Guibas, L.J.: Image co-segmentation via consistent functional maps. In: ICCV. (2013)
49. Chen, X., Shrivastava, A., Gupta, A.: Enriching visual knowledge bases via object discovery and segmentation. In: CVPR. (2014)
50. Jerripothula, K.R., Cai, J., Meng, F., Yuan, J.: Automatic image co-segmentation using geometric mean saliency. In: ICIP. (2014)
51. Jerripothula, K.R., Cai, J., Yuan, J.: Image co-segmentation via saliency co-fusion. IEEE Transactions on Multimedia **18**(9) (2016) 1896–1909