# Image Cosegmentation via Multi-task Learning

Qiang Zhang, Jiayu Zhou, Yilin Wang, Jieping Ye, Baoxin Li
qzhang53,jiayu.zhou,ywang370,jieping.ye,baoxin.li@asu.edu

Computer Science and Engineering
Arizona State Uni., Tempe, AZ, USA

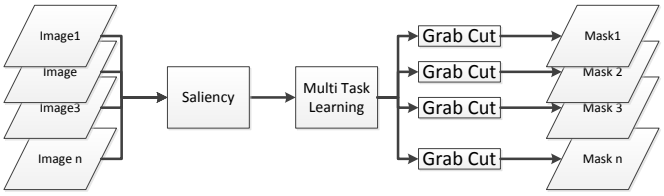


Figure 1: The overview of the proposed algorithm. We first extract the saliency map for the input images; according to the saliency map, we pick the seed regions to initialize the multi-task learning algorithm; and finally according to the output of multi-task learning algorithm, we use grab cut to obtain the final segmentation result.

## 1 Motivation

Image segmentation has been studied in computer vision for many years and yet it remains a challenging task. One major difficulty arises from the diversity of the foreground, which often results in ambiguity of background-foreground separation, especially when prior knowledge is missing. To overcome this difficulty, cosegmentation methods were proposed, where a set of images sharing some common foreground objects are segmented simultaneously. Different models have been employed for exploring such a prior of common foreground. In this paper, we propose to formulate the image cosegmentaion problem using a multi-task learning framework, where segmentation of each image is viewed as one task and the prior of shared foreground is modeled via the intrinsic relatedness among the tasks. Compared with other existing methods, the proposed approach is able to simultaneously segment more than two images with relatively low computational cost. The proposed formulation, with three different embodiments, is evaluated on two benchmark datasets, the CMU iCoseg dataset and the MSRC dataset, with comparison to leading existing methods. Experimental results demonstrate the effectiveness of the proposed method.

## 2 Proposed Method

An overview of the proposed method is illustrated in Fig 1. In experiments of this paper, we first over-segment the images into superpixels and then use them as basic units for subsequence processing. For obtaining the superpixels, we use SLIC and set the number of superpixels for each image to 200. For notations, we use $\mathbf{X}_i^j$ to represent the descriptor of the $j_{th}$ superpixel in $i_{th}$ image and $y_i^j$ as its label.

**Feature Extraction**: for each superpixel, we extract the feature according to [15], which includes geometry measurements, color, texture and edges. The similarity measure of the superpixels is one of the most important component for image segmentation. For image cosegmentation, we need not only the similarities measure of the superpixels within each image, but also the similarities measure of superpixels cross different images. For the superpixels within each image, high similarity score is assigned to superpixels which are both spatially close and feature-wise similar. For the similarities of the superpixels cross images, we use nearest neighbor to find their correspondences.

$$A(i,j;p,q) = K(\mathbf{X}_i^j, \mathbf{X}_p^q) \times e^{\frac{-|loc(\mathbf{X}_i^j) - loc(\mathbf{X}_p^q)|_2^2}{2\sigma}} \text{ if } i = p \quad (1)$$

$$A(i,j;p,q) = K(\mathbf{X}_i^j, \mathbf{X}_p^q) \times \text{KNN}(i,j;p,q) \text{ if } i \neq p \quad (2)$$

**Visual Cosaliency**: recently visual cosaliency was proposed and utilized to initialize the image coegmentation algorithm. In the proposed cosaliency, a superpixel is cosalient, if it is not only salient in the corresponding image but also similar to salient superpixels of other images. After computing the cosaliency score $s_i^j$, we label the top 20% of the salient superpixel as the foreground and the bottom 70% ones as background to initialize the image cosegmentation algorithm.

$$s_i^j(t+1) = (1-\alpha)\hat{s}_i^j + \alpha \sum_{p,q:s_p^q(t)\geq\tau} s_p^q(t) \times A(i,j;p,q) \quad (3)$$

| | |
|---|---|
| Mean | 88.67% |
| $\ell_{2,1}$ | 87.81% |
| Low | 88.33% |

(a)

| | |
|---|---|
| Mean | 80.58% |
| $\ell_{2,1}$ | 80.87% |
| Low | 81.20% |

(b)

Table 1: (a) The result on iCoseg dataset. (b) The result on MSRC dataset.

Figure 2: Example of image segmentation on iCoseg dataset and MSRC dataset, where the green contour shows the segmentation results.

**Multi-task learning**: we formulate image cosegmentation as a multi-task learning problem, where the segmentation of each image is viewed as one task. Naturally, in this formulation the prior that common foreground objects are shared among the images is assumed to be captured by the intrinsic relatedness among the tasks. For developing a solution under this formulation, we focus on regularization-based modeling, due to its flexibility in incorporating existing computational models and supporting various assumptions of task relatedness. Especially we consider the following three multi-task learning assumptions:

**1) the task parameters are drawn from the same distribution ("mean").**

$$\{\mathbf{W}_i\} : \arg\min_{\{\mathbf{W}_i\}} f(\{\mathbf{W}_i\}) + \lambda \|\mathbf{W}_i - 1/N \sum_{j=1}^{N} \mathbf{W}_j\|_2^2. \quad (4)$$

**2) the models of the tasks share a common low-rank subspace ("low").**

$$\{\mathbf{W}_i\} : \arg\min_{\{\mathbf{W}_i\}} f(\{\mathbf{W}_i\}) + \lambda \|\mathbf{W}\|_* \quad (5)$$

**3) the models of the tasks share the same subset of features ("$\ell_{2,1}$").**

$$\{\mathbf{W}_i\} : \arg\min_{\{\mathbf{W}_i\}} f(\{\mathbf{W}_i\}) + \lambda \|\mathbf{W}\|_{2,1} \quad (6)$$

After we find the classifiers with the multi-task learning methods, we apply the classifiers to each superpixel of the images. The classifiers return a score within $[0,1]$ via logistic function. We then label a superpixel as background if its response is smaller than 80% of the mean response of all of the superpixels of all images, which will be used for initialization of Grabcut algorithm to obtain the final segmentation.

## 3 Experiment

We evaluate the proposed method on two widely-used datasets: CMU iCoseg (37 sets of images, 4 to 41 images per class) and MSRC (14 sets of images, around 30 images for each set). We compared with several existing methods, some of them being the current state-of-art. For performance metric, we compute the accuracy of the segmentation result over the manually labeled mask, which includes both foreground and background $p = \frac{\text{true postive+true negative}}{\text{area of image}}$.