Unsupervised co-segmentation through region matching

Jose C. Rubio[1], Joan Serrat[1], Antonio López[1], Nikos Paragios[2,3,4]

[1]Computer Vision Center & Dept. Comp. Science, Universitat Autònoma de Barcelona, Cerdanyola, Spain. *

[2]Center for Visual Computing, Ecole Centrale de Paris, France.

[3]Université Paris-Est, LIGM (UMR CNRS), Center for Visual Computing, Ecole des Ponts ParisTech, France.

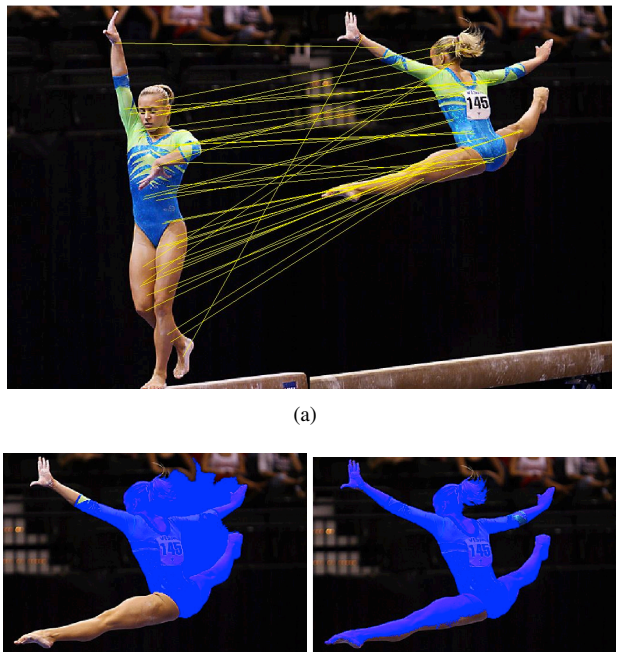
[4]Equipe Galen, INRIA Saclay, Ile-de-France, France.

Abstract

Co-segmentation is defined as jointly partitioning multiple images depicting the same or similar object, into foreground and background. Our method consists of a multiple-scale multiple-image generative model, which jointly estimates the foreground and background appearance distributions from several images, in a non-supervised manner. In contrast to other co-segmentation methods, our approach does not require the images to have similar foregrounds and different backgrounds to function properly. Region matching is applied to exploit inter-image information by establishing correspondences between the common objects that appear in the scene. Moreover, computing many-to-many associations of regions allow further applications, like recognition of object parts across images. We report results on iCoseg, a challenging dataset that presents extreme variability in camera viewpoint, illumination and object deformations and poses. We also show that our method is robust against large intra-class variability in the MSRC database.

1. Introduction

Bottom-up segmentation of generic images is a long standing goal in computer vision for its many potential applications. It is a highly unconstrained and ill-posed problem which has given rise to a multitude of approaches. Co-segmentation is a recent approach to cope with this lack of constraints. Given two or more images showing the same object, or similar instances of the same object class, the goal is to partition them into foreground (object) and background regions under the assumption that the background changes significantly while the foreground does not. Co-segmentation methods leverage this fact in order to determine what is the foreground region. Note that this is a chicken and egg problem: the aim is to compare something —the two foregrounds— that is still unknown. As difficult as it may appear, it is a especially appealing approach because, in its purest version, it only requires providing an additional image containing the same or a similar instance of the target object. Several applications of co-segmentation have been explored in the literature. One consists of increasing the performance of image retrieval by removing the background from the image similarity metric, thus focusing on the object which is sought [15]. Also, automatically segmenting the instances of an object in a collection of pictures would allow to create visual summaries [2]. A few



(a)

(b) (c)

Figure 1: In (a) yellow lines show region matching results for the foreground area of two images. In (b) the blue-colored pixels show the results of the foreground objectness-based initialization. In (c), our results.

*This work was supported by the Spanish Ministry of Education and Science under Project TRA2011-29454-C03-01, TIN2011-29494-C03-02, and the Research Program Consolider Ingenio 2010: MIPRCV (CSD2007–00018)

more specific applications are the segmentation of neuron cells from electron microscopy images sequences [18], the reconstruction of 3D models of individuals (with user interaction) [12] and recognition of people wearing the same clothes [7].

In this paper we describe a new approach to the co-segmentation problem which has several unique characteristics (see section 1.2). Based on matching superpixels resulting from an oversegmentation algorithm, it not only produces foreground/background partitions but also relates their regions. We believe that this will allow to extend the applications of co-segmentation to those needing the correspondence between foreground pixels, contours or regions, like parts-based recognition or 3D reconstruction. Another distinctive feature is that we model both the hypothesized foreground and background appearance distributions separately. Thus, the assumption of similar foreground and changing background is replaced by image-specific distributions of foreground and background. Provided that they can be well estimated, the co-segmentation may succeed even though the background does not change much.

1.1. Previous work

In the seminal work by Rother *et al.* [15], the problem was posed as labeling pixels as foreground or background through energy minimization. The energy included a key term to measure the L1-norm dissimilarity between the unnormalized foreground histograms and another pairwise term regularizing the solution in both images.

Subsequent methods [15, 16, 13] compared foreground color histograms in different ways, succeeding on relatively simple image pairs. However, color histograms are clearly dependent on lighting conditions and also on the foreground scale since they are not normalized. Non surprisingly, the most recent works employ additional features, such as SIFT and texture descriptors [14], saliency [4], and Gabor filters [9].

Maybe the distinctive trait of each method is how does it build the partition hypothesis and perform the foreground comparison, that is, how to cast the co-segmentation problem into some solvable formalism. The dominant approach is minimization of an energy function equivalent to MAP estimation on a MRF [15, 16, 4, 2] but other original approaches have been tried like the minimization of a quadratic pseudoboolean function [14] or max-flow min-cut optimization [9]. More interesting, Vicente *et al.* [17] generate a large number of candidate segmentations for the two images by means of a variation of Min-Cuts. Then a Random forest regressor, trained with many pairs of ground-truth segmentations, scores each pair of segmentations, one for each image. An exact A* search finds the pair of segmentation proposals with the highest score. This is one of the few automatic methods reporting results on iCoseg, a challenging benchmarking dataset.

Closely related to co-segmentation is the problem dubbed as co-clustering. The goal is similar though the approach is not. Given two or more images and their oversegmentations, the aim is to group the regions in each image into two or more clusters, each corresponding to an object of interest. One difference with respect to co-segmentation is that co-clustering concerns regions, not pixels. Glasner *et al.* [8] perform this clustering by comparing the color and shape of groups of regions. They are thus able to co-segment two or more images with similar backgrounds provided that the foreground shape is roughly the same, like in nearby frames of a video sequence. They pose co-clustering as a quadratic semi-assignment problem which is solved by linear programming relaxation, like in [18]. Joulin *et al.* address co-segmentation of two or more images by means of unsupervised discriminative clustering. Their elegant formulation ends up in a relaxed convex optimization.

All these works provide original formulations and successful results for automatic co-segmentation. But only a few of them [17, 10] go beyond the relatively simple image pairs of the first papers ('banana', 'bear', 'dog', ... on varying backgrounds), and focus on the challenging datasets iCoseg and MSRC. iCoseg contains a varying number of images of the same object instance under very different viewpoints and illumination, articulated or deformable objects like people, complex backgrounds and occlusions. MSRC contains images of different objects belonging to the same class, with varying aspect.

1.2. Goal

The novelty of our work is a new automatic co-segmentation method which exhibits the following characteristics:

- Fully unsupervised, meaning that there is no need of training with ground-truth segmentations of images from the same or from other classes (like in [16]).
- Able to work with more than two images, a novelty just explored in recent papers [10, 14, 8]. This means that not only the formulation is more general but that the method has to scale well with the number of images.
- The method not only produces a foreground / background partition of the images but also computes many-to-one and one-to-many associations (correspondences) among regions from different images. These regions may constitute object parts and thus co-segmentation would allow further applications.
- It is comparable with state of the art non-supervised *and* supervised methods on the benchmark datasets iCoseg and MSRC. On this regard, we take as reference the very recent works by Vicente *et al.* [17] and Joulin *et al.* [10] for the reasons mentioned previously.

- Performing well in the difficult case of similar backgrounds, overcoming the constraint associated with the first co-segmentation methods, as explained above.

The reminder of this paper is organized as follows: In section 2, we formulate the co-segmentation problem as an unsupervised binary-class labeling of a set of images depicting the same object instance. We start proposing a multi-image representation, and define the problem in terms of an energy minimization. In section 3, we extend the scene representation to include several segmentation proposals to capture the image elements at different scales, and we present a generative appearance model of the scene, trained at testing time. In section 4, we show that spectral matching can be used to establish correspondences between image regions, by exploiting the underlying information of region arrangements. Section 5 presents the experimental set-up and results on standard databases, while the last section concludes the paper.

2. Co-segmentation Formulation

Let us consider a set of images $I = \{I_1, I_2, ..., I_n\}$ containing an instance of the object of interest. Let us also consider that the images have been partitioned based on visual appearance, by a segmentation algorithm such as mean-shift [5]. We propose a two-layered MRF composed of region nodes and pixel nodes. The indexing of nodes is denoted by $\mathcal{V} \in \mathcal{V}_r \cup \mathcal{V}_p$, where the sets $\mathcal{V}_r$ and $\mathcal{V}_p$ correspond to regions and pixel nodes respectively. Slightly abusing notation, we write $\mathcal{V}_r(k)$ to denote the regions of image k, and $\mathcal{V}_p(k)$ to refer to the pixels. The MRF comprises a vector of boolean random variables $\mathbf{X} = (X_i)_{i\in\mathcal{V}_r} \cup (X_j)_{j\in\mathcal{V}_p}$. The varsables can take two values: $X = 0$ indicates background and $X = 1$ indicates foreground. Our goal is to infer a consistent labeling of regions and pixels that better separates the foreground and background areas of the image set. We state the problem as the minimization of the following energy function:

$$E(\mathbf{X}) = \lambda_1 E^{pixel} + \lambda_2 E^{region} + E^{scale} + E^{matching} \quad (1)$$

The first two components E^{pixel} and E^{region} are unary potentials encoding the likelihood of pixels and regions belonging to foreground or background, weighted by parameters λ_1 and λ_2. The E^{scale} term enforces a consistent labeling of pixels and the region they belong to. The last term $E^{matching}$ encourages coherent inter-image labeling of regions. The next sections detail the formulation of these terms. Figure 2. shows an example of MRF.

3. Generative Foreground/Background Model

We first compute several segmentation proposals at different scales in order to have a rich description of the scene. Then, an iterative algorithm infers the appearance likelihood distribution of foreground and background. The peculiarity of our framework is that the distributions are trained at testing time using a rough estimate of the image foreground/background labeling, instead of ground-truth annotations.

3.1. Multi-scale Segmentation Pool

An over-segmented dictionary $\mathcal{R}$ of regions is generated using mean-shift with different sets of parameters, over every image in the set I. The original region set $\mathcal{V}_r$ is redefined to include every region of the dictionary, for all images: $\mathcal{V}_r = \mathcal{R}_k, \forall I_k \in I$. Our model comprehends pixels as well as regions. Each pixel has as many parent regions as levels of segmentations computed to build the dictionary (See Figure 2). To encourage a coherent labeling between regions and pixels we introduce the first energy component E^{scale}, as

$$E^{scale}(\mathbf{X}) = \sum_{(i,j)\in\Delta} \eta(1 - \delta(X_i, X_j)), \quad (2)$$

where the cost η penalizes pairs of pixel and region nodes with different labels. The function δ is the Dirac delta function, and the set Δ contains the indexes of every pair of overlapping regions and pixels.

While pixel labeling helps to overcome errors propagated from the mean-shift segmentations (providing finer labeling atoms), the region level enforces spatial grouping. Moreover, multiple segmentations capture the image semantics at different scales, making the inter-image region matching robust against scale variations.

3.2. Pixel and Region Potentials

Defining the likelihood of a pixel/region belonging to the foreground or the background in a unsupervised framework is a challenging task. A priori, there is no information available about such distributions. However, analogously to [17], we assume without loss of generality that the image foreground is an object. Therefore, we can use the algorithm of [1] as a measure of objectness. Note that the objectness measure is applied out-of-the-box, without re-training with the databases used in the experiments. This is important because we want our method to be free of ground-truth segmentations, or ground-truth class labeling. The method of [1] outputs a prior of the object location as the probability of covering an object with a sampled window. We sample 10^4 bounding boxes, and calculate the probability of a pixel belonging to an object by simply averaging the score of every bounding box containing that pixel. Then we extend these scores to the regions, by averaging the probabilities of every pixel contained in each of the regions.

One of the typical requirements for performing co-segmentation is that the appearance of the foreground dif-

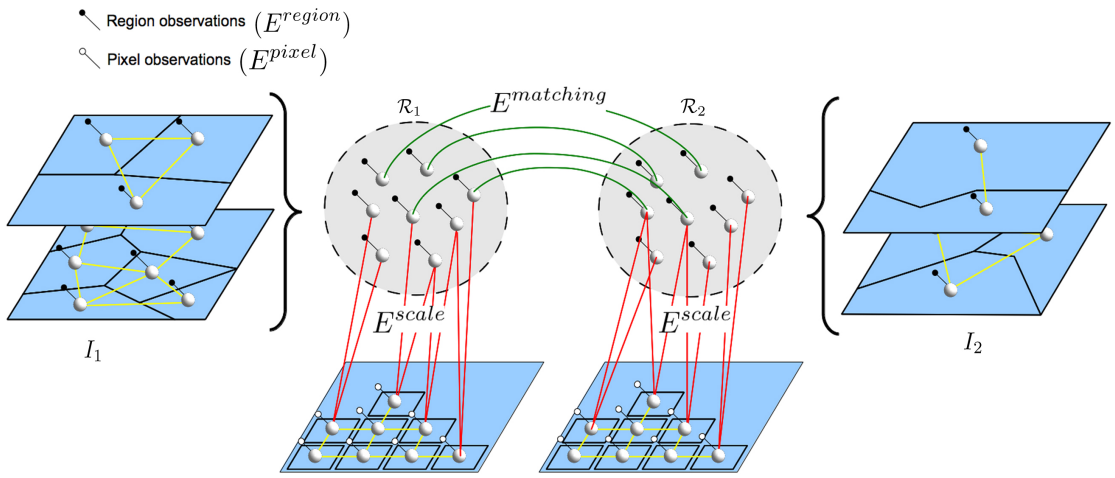


Figure 2: Markov Random Field of the multi-scale multi-image model (two images), illustrating the energy components. In the left and right sides, it is shown the pool of proposal segmentations (two scales) for images I_1 and I_2. The two big gray circles represent the dictionaries of regions $\mathcal{R}_1$ and $\mathcal{R}_2$. The small white and black circles denote the singleton potentials for regions and pixel nodes. The red vertical edges (Δ) connect pixel nodes with regions from the image dictionaries. The green horizontal lines ($\mathcal{E}$) show examples of region correspondences. The yellow intra-image edges between pixels and regions denote optional smoothing potentials.

fers from that of the background to a certain extent. We propose a generative image model, on which the objectness measure plays a guiding role to iteratively infer both distributions. The inference is based on the premise that the objectness-based initial distribution resembles the foreground and is distinct to the background. Note that even if this may not be always true for every image, it is very likely that it remains true if we jointly model the distribution from several images. Figure 1.(b) shows an example of an objectness initialization.

The distributions are constructed with simple features. We use the RGB color of the image pixels and a texture descriptor extracted from every region of the set $\mathcal{R}$. In our implementation we use Gaussian Mixture Models (GMM) to estimate pixel color distributions. The foreground and background GMMs are denoted as $\mathcal{H}^f$ and $\mathcal{H}^b$ respectively. We also train a texture-based appearance model of the image regions using a texture descriptor (Local Binary Pattern). However, in this case, a parameter estimation for GMMs with high dimensionality (50 bins) requires a large amount of training data and computation time. Instead, a linear SVM classifier $\mathcal{F}$ is trained over the texture descriptors of the foreground and background estimation.

The algorithm starts by initializing the labels of pixels and regions using the output of the objectness measure, and estimating the color and texture distribution of the background and foreground from this first rough labeling. We feed our unary potentials by querying the learnt distributions, and optimize the energy function of Eq. 1 to obtain a new labeling. Then, we iteratively update the distributions from the last output labeling until reaching a maximum number of iterations, or a convergence criteria. The procedure is detailed in Algorithm. 1. Constructing independent distributions for texture and color makes the model robust against difficult cases where the foreground and background have a similar appearance. When one of the features (color, texture) is not discriminative enough, we rely on the other to forbid one distribution to *leak* into the other.

A weakness of such an iterative approach is the initial seeding. A poor initialization results in ill-formed distributions with spurious samples that may bias the foreground model towards the background, and vice versa. In practice, what we observe is that one of the distributions slowly expands with every iteration, and quickly covers every pixel of the image. In order to make the model robust to poor initializations, we build as many appearance models as images in I in such a way that the distribution corresponding to image I_k is constructed with the information of every image except I_k. Following this approach, the wrong training samples will not contribute with a high probability when the same samples are used at testing time to query the distribution. We extend the previous formulation to denote the two GMMs of a specific image k as $\mathcal{H}^f_k$ and $\mathcal{H}^b_k$. This also applies to the texture classifier of image k, now denoted as $\mathcal{F}_k$. Given the set of n images, the singleton potential for the regions is formulated below, as the logarithm of the probability estimate returned by the texture classifier.

$$E^{region}(\mathbf{X}) = \sum_{k}^{n} \sum_{i \in \mathcal{V}_r(k)} -log(\hat{P}^f_k(T_i)X_i + \hat{P}^b_k(T_i)\overline{X}_i), \tag{3}$$

The probability $\hat{P}^f_k(T_i)$ is the estimate for the *foreground label* predicted by the classifier $\mathcal{F}_k$ on the texture descriptor T_i of region i. The term $\hat{P}^b_k(T_i)$ is the estimate for the

background label on the same texture descriptor.

The singleton potential of a pixel node is the resulting cost of applying the logarithm to the color likelihood distribution $\mathcal{H}$. Formally,

$$E^{pixel}(\mathbf{X}) = \sum_{k}^{n} \sum_{j \in \mathcal{V}_p(k)} -log(P(C_j|\mathcal{H}_k^f)X_j + P(C_j|\mathcal{H}_k^b)\overline{X}_j), \tag{4}$$

where C_j is the color (e.g. RGB value) of pixel j. The term $\mathcal{H}_k^f$ refers to a gaussian mixture model trained on the foreground pixels of every image except I_k, and $\mathcal{H}_k^b$ is the analogous model for the background.

Algorithm 1 Iterative Foreground/Background Modeling

1: Initialize $\mathbf{X}_i, \mathbf{X}_j \leftarrow$ objectness, $\forall i \in \mathcal{V}_r, \forall j \in \mathcal{V}_p$.
2: **repeat**
3: Estimate $\mathcal{H}_k^f \leftarrow GMM(\mathbf{X}_j = 1), \forall I_k \in I$
4: Estimate $\mathcal{H}_k^b \leftarrow GMM(\mathbf{X}_j = 0), \forall I_k \in I$
5: Train SVM $\mathcal{F}_k \leftarrow \mathbf{X}_i, \forall I_k \in I$
6: $\mathbf{X}^* \leftarrow argmin_{\mathbf{X}} E(\mathbf{X})$
7: Update labels $\mathbf{X}_i, \mathbf{X}_j \leftarrow \mathbf{X}^*$
8: **until** convergence

4. Region Matching

A key aspect when tackling the co-segmentation problem is the exploitation of the inter-image information. For challenging cases in which objects are deformable and change considerably in terms of viewpoint and pose, it is difficult to leverage the spatial distribution of the regions in order to find correspondences. Usually, matching methods establish geometric constraints on the image structure by preserving a distance measure between nodes embedded in a Euclidean space. One major drawback of this approach is the restriction to a near-rigid or near-isometric assignment, which results in poor performance when there are large variations in the node arrangements. We overcome this limitation by relying in the statistical properties of the graph of regions, by applying commute times as a distance between pairs of matching regions.

Let (r_i, r_j) be the indexes of two arbitrary regions from the dictionary $\mathcal{V}_r$. The distance between regions is defined as

$$D(r_i, r_j) = \alpha d(C_i, C_j) + (1 - \alpha) d(S_i, S_j), \tag{5}$$

where C denotes the RGB color of the image region as the mean color of the pixels contained in it. In the second term, S refers to the SIFT descriptor extracted from the image regions, obtained by computing a dense SIFT on every pixel of a region using a 16 by 16 patch, and clustering them in 8 bins. The function d is a χ^2- distance measure, and α is a weight expressing the influence of feature similarity.

The structure of regions within an image is represented as a graph of regions, with its adjacency matrix defined as:

$$\Omega(r_i, r_j) = \begin{cases} D(r_i, r_j) & \text{if } r_i \text{ shares a boundary with } r_j \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

Commute times have been recently used in [3] to characterize the layout of a graph, proving to be stable against structural variations of the scene. The commute time matrix between regions can be efficiently computed from the spectrum of the normalized Laplacian of the adjacency graph:

$$CT(r_i, r_j) = vol \sum_{k=2}^{N} \frac{1}{\lambda_k} \left(\frac{\pi_k(r_i)}{\sqrt{d_i}} - \frac{\pi_k(r_j)}{\sqrt{d_j}} \right)^2 \tag{7}$$

where $vol = \sum_{k=1}^{N} d_k$, and d_k is the degree of node k. The terms π_k and λ_k denote the k^{th} eigenvalue and eigenvector of the graph Laplacian.

We denote every possible correspondence a between one region in image I_1 and another region in image I_2 as $a = (r_i, r_j) \in I_1 \times I_2$. The matching score of those correspondences is defined by the matrix M, where,

- $M(a, a)$ denotes the affinity of individual assignments given by the distance between regions defined in Eq. 5. Given a correspondence $a = (r_i, r_j)$,

$$M(a, a) = D(r_i, r_j) \tag{8}$$

- $M(a, b)$ defines how well a pair of region correspondences match. In our case, we use this term to preserve a commute time distance between pairs of regions in correspondence. Given a pair of correspondences $a = (r_i, r_j), b = (r_k, r_l)$,

$$M(a, b) = \frac{|CT(r_i, r_j) - CT(r_k, r_l)|}{CT(r_i, r_j) + CT(r_k, r_l)} \tag{9}$$

As stated in [6], the matching problem reduces to find the set of region correspondences $(r_i, r_j) \in \mathcal{E}$ that maximizes the score M. If we represent the set of possible correspondences as a vector of indicator variables, such that $y(a) = 1$ if $a \in \mathcal{E}$, and zero otherwise, the matching problem can be formulated as the following optimization:

$$y^* = argmax(y^T M y) \tag{10}$$

We use the algorithm of [6] to optimize the above objective, and define a new set of corresponding regions $\mathcal{E} = \{a | y^*(a) = 1\}$, matching every possible pair of images of the input set. Finally, we introduce the last energy term, $E^{matching}$, which imposes a penalty on corresponding regions with different labels. We can write it analogously to Eq. 2, as

$$E^{matching}(\mathbf{X}) = \sum_{(i,j) \in \mathcal{E}} \theta(1 - \delta(X_i, X_j)), \tag{11}$$

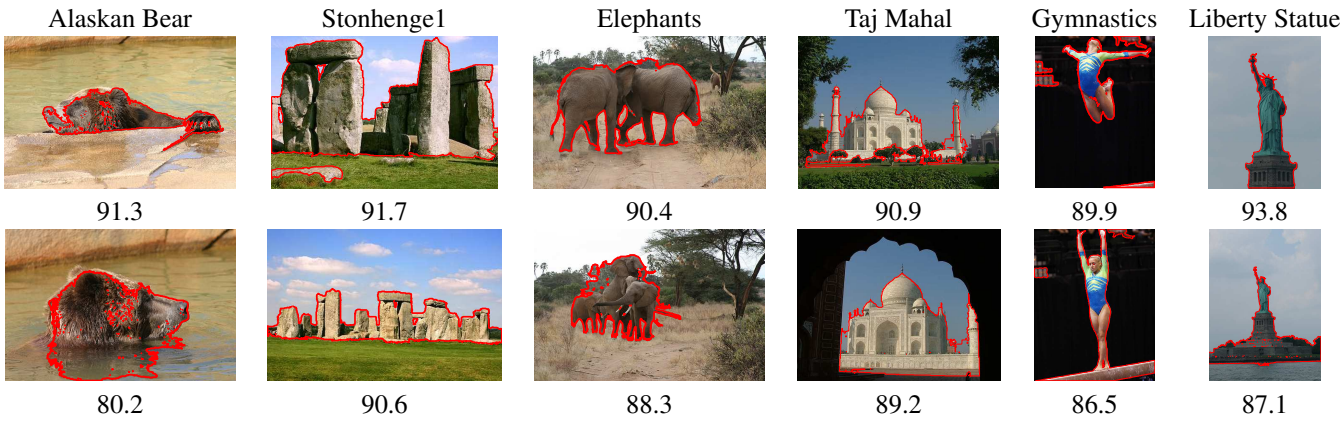


Figure 3: Example results on the iCoseg dataset. A red line separates foreground and background areas.

iCoseg	Ours	[10]	[17]	uniform	obj.
Alaskan bear	**86.4**	74.8	90.0	79.0	79.5
Red Sox Players	**90.5**	73.0	90.9	86.8	77.2
Stonehenge1	**87.3**	56.6	63.3	88.8	85.5
Stonehenge2	**88.4**	86.0	88.8	68.4	70.0
Liverpool FC	**82.6**	76.4	87.5	82.9	85.0
Ferrari	84.3	**85.0**	89.9	73.9	78.0
Taj Mahal	**88.7**	73.7	91.1	83.4	74.9
Elephants	**75.0**	70.1	43.1	83.5	80.6
Pandas	60.0	**84.0**	92.7	68.7	81.3
Kite	**89.8**	87.0	90.3	76.0	77.3
Kite panda	**78.3**	73.2	90.2	62.0	78.4
Gymnastics	87.1	**90.9**	91.7	62.7	75.8
Skating	76.8	**82.1**	77.5	73.7	72.9
Hot Balloons	**89.0**	85.2	90.1	78.2	84.1
Liberty Statue	**91.6**	90.6	93.8	64.4	79.4
Brown Bear	**80.4**	74.0	95.3	82.2	78.1
mean accuracy	83.9	78.9	85.3	75.9	78.6

Table 1: Bold numbers represent classes in which our method's performance overcomes the non-supervised competitor [10]. In bold red, the scores for classes in which our method outperforms the state-of-the-art supervised method. In the second column the results as reported in [17].

by noting that the penalty is equal to 0 when both corresponding regions belong to the foreground or both to the background, and θ otherwise.

5. Qualitative and Quantitative results

We evaluate our method with three different experiments. We report results on the iCoseg and MSRC datasets, and we illustrate the application of co-segmentation by matching, with an example of part-based correspondence.

We present qualitative and quantitative results of our algorithm. The segmentation accuracy of a given image is measured by computing the ratio of correctly labeled pixels of foreground and background with respect to the total number of pixels, like in [17].

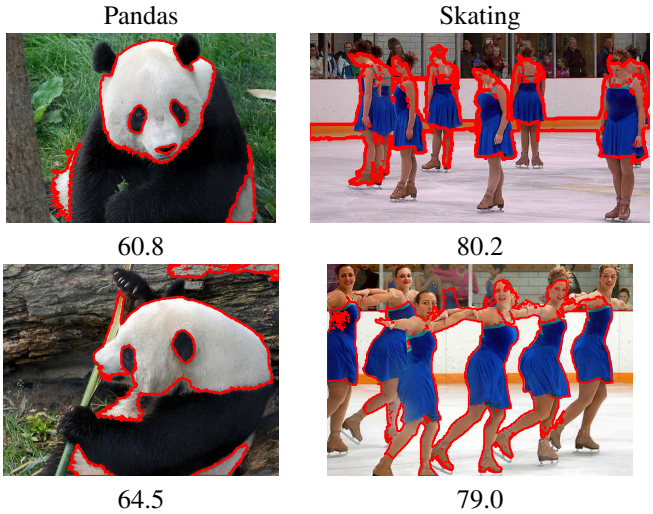


Figure 4: Examples of cases where the method fails in classes *Panda* and *Skating*. In the case of the panda, the objectness initialization leads the foreground to a local minima on the white fur. The images of the *Skating* class present complex compositions of objects.

5.1. Experimental set-up

The sub-modularity of the pairwise potentials is assured, since we only apply a cost on the variable configurations $[X_i \neq X_j]$. This lets us optimize the objective using graph-cuts but, in principle, any other optimization algorithm could be used as well. We choose graph-cuts because it provides a good trade-off between optimization speed and low energy bound. The number of components in the GMMs is automatically determined using the center-based clustering algorithm proposed in [11]. The last detail remaining is the value of the penalty scalars (θ, η) and unary weights (λ_1, λ_2). Since we want to avoid learning these parameters from training data, we adopt a conservative approach to set them: we set both penalties to the minimum integer cost 1, and we uniformly assign the same weight $\lambda = \lambda_1, \lambda_2$ to both singleton potentials. At the end, our method only depends on three parameters: λ, α, and the maximum number

of iterations. It is worth to mention that only the λ value has an important influence on the algorithm performance. The stopping condition of the iterative process depends on the ratio of pixels that switched label since the last iteration. If this percentage is less than 2.5%, the algorithm stops. We use three different levels of scale segmentations, with the same mean-shift parameters for every image. We set the parameter α which weights the contribution of color and SIFT in the distance measure to 0.5. The parameter λ to scale the appearance potential is set to 2.75.

It is very common in the literature to apply a smoothing on the label values using an extra pairwise term between neighbor variables. We leave it as an optional energy component, because it proved to be not much influential in the performance of the algorithm, and we avoid setting an extra penalty parameter.

5.2. iCoseg database

The iCoseg database was introduced in [2]. It contains 643 images divided into 38 classes with hand-labelled pixel-level segmentation ground-truth. Each class is composed of approximately 17 images. For this experiment, and for the sake of comparison, we use the same sub-set of 16 classes reported in [17]. We simultaneously co-segment groups of (at most) 10 images, and average the results for each of the groups. The images of each group are randomly selected, to avoid unfair grouping of affine-looking images.

In Table 1, we compare the performance of our method with a recent non-supervised method [10]. That is, a method that does not require ground-truth segmentations of object instances. Our results are on line with state-of-the-art algorithms such [17] (third column), which trains a pairwise energy from groundtruth segmentations of pairs of objects, tested against new groups of images.

The fourth column shows results with a uniform segmentation: best error rate of full (all ones) and empty (all zeros) segmentations. The last column contains the objectness-based initialization results. The bold red figures show classes in which our method outperforms [17]. This is mainly due to the high similarity on the image background (Elephant and Stonehenge), for which our method performs better because it estimates both foreground and background distributions. On average, our result for all 16 classes is slightly below [17] (just -1.4%).

The *Pandas* class performs the worst because an incorrect objectness initialization in the majority of its images keeps the foreground distribution *trapped* inside the white fur patches of the panda (See Figure 4). If the object presents a high color variability within the same instance, a correct initialization of the first foreground estimate is key to achieve satisfactory results. The *Skating* class is difficult to segment due to the complexity of the object. Again, the appearance variability between the color of the skaters' costume and the body parts, makes the foreground distribution fall in a local minima covering only the costume. The skaters' legs and heads are labeled as background.

Figure 5: Example results on the MSRC dataset. The images show a high color variability between foregrounds of the same class, in *Horse* and *Cars (back)*. The background of the *plane* class does not significantly change.

MSRC	images	Ours	Joulin et al.[10]	uniform
Cars (front)	6	65.9	**87.65**	64.0
Cars (back)	6	52.4	**85.1**	71.3
Face	30	76.3	**84.3**	60.4
Cow	30	80.1	**81.6**	66.3
Horse	30	74.9	**80.1**	68.6
Cat	30	**77.1**	74.4	59.2
Plane	30	**77.0**	73.8	75.9
Bike	30	62.4	**63.3**	59.0

Table 2: Segmentation accuracy for the MSRC dataset and Weizzman horses.

5.3. Objects with variate appearances

The MSRC database depicts images of different instances of the same class. This contradicts one of the main hypotheses of our method, which strongly relies on a unique aspect of the foreground object. For instance, some classes show objects with variate colors and textures (e.g. Cars). We show that our method performs reasonably well even though this assumption does not hold.

Table 2. shows comparative results on the MSRC dataset and Weizman Horses dataset. In comparison to [10], our method outperforms the reported results when the background hardly changes, such as the *plane* class. As pointed out in [10], plane images have a similar background (the airport) that makes the task of separating foreground and background harder. Our algorithm does not suffer from this limitation, as long as foreground and background do not look alike.

Cars perform poorly due to the small number of images available in the database (6), and the large intra-class variability, especially regarding color. Figure 5. shows an example of this. Our method tends to identify windows and light-beams as the common foreground region.

5.4. Part-based recognition

Obtaining region correspondences within a co-segmentation framework is very useful for understanding the semantics of a scene. In our third experiment we use the region matching output to identify parts of the common objects co-segmented. We simply gather the regions selected as foreground by our co-segmentation algorithm, and select the correspondences with the highest spectral matching scores. Figure 6. shows the corresponding regions of four images of the *Kendo* class from the iCoseg database. The images are paired in two sets to show the part-recognition of each of the fighters. The heads and regions close to the head present the higher matching scores, being the only regions with discriminative features. In Figure 6. (e,f), only the upper part of the body finds matching candidates scoring over the threshold. In (h), the whole body of the fighter is correctly matched.

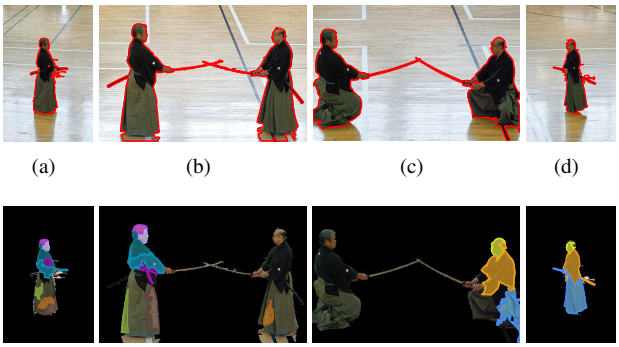


Figure 6: In (a,b,c,d), the output of the co-segmentation. The pair (e,f) shows an example of matching, and (g,h) another. Each color represents a correspondence between a pair of foreground regions.

6. Conclusions

We have proposed a multi-scale multi-image representation that is able to model complex scenes containing several objects. A non-supervised iterative algorithm is presented, that is able to separate foreground and background by modeling both appearance distributions on pixels and regions. We show that is possible to take advantage of an explicit matching of regions, that increases the consistency of the foreground and background models across images. Our approach has shown to be robust against deformable objects as well as changes on object poses and camera viewpoint. We also overcome the limitation of other recent methods, which require background dissimilarity among the input images.

Our algorithm shows competitive results with state-of-the-art methods, and outperforms recent non-supervised co-segmentation approaches. It has shown good performance on two types of databases, one (iCoseg) contains the same object instance per class, while the other (MSRC) contains objects with varied appearances within the same class. One of the main advantages of our method is that it does not require different backgrounds in the input images. Other advantage is that the performance improves with the number of images available. We also present a prove of concept to illustrate the use of co-segmentation as a starting point to perform part-based recognition.

References

[1] B. Alexe, T. Deselaers, and V. Ferrari. What is an object? In *CVPR*, 2010. 3

[2] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen. icoseg: Interactive co-segmentation with intelligent scribble guidance. In *CVPR*, 2010. 1, 2, 7

[3] R. Behmo, N. Paragios, and V. Prinet. Graph commute times for image representation. In *CVPR*, 2008. 5

[4] K.-Y. Chang, T.-L. Liu, and S.-H. Lai. From co-saliency to co-segmentation: An efficient and fully unsupervised energy minimization model. In *CVPR*, 2011. 2

[5] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2002. 3

[6] O. Duchenne, F. Bach, I.-S. Kweon, and J. Ponce. A tensor-based algorithm for high-order graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2011. 5

[7] A. C. Gallagher and T. Chen. Clothing cosegmentation for recognizing people. In *CVPR*, 2008. 2

[8] D. Glasner, S. N. P. Vitaladevuni, and R. Basri. Contour-based joint clustering of multiple segmentations. In *CVPR*, 2011. 2

[9] D. S. Hochbaum and V. Singh. An efficient algorithm for co-segmentation. In *ICCV*, 2009. 2

[10] A. Joulin, F. R. Bach, and J. Ponce. Discriminative clustering for image co-segmentation. In *CVPR*, 2010. 2, 6, 7

[11] N. Komodakis, N. Paragios, and G. Tziritas. Clustering via lp-based stabilities. In *NIPS*, 2008. 6

[12] A. Kowdle, D. Batra, W. Chen, and T. Chen. imodel: interactive co-segmentation for object of interest 3d modeling. In *ECCV*, 2010. 2

[13] L. Mukherjee, V. Singh, and C. R. Dyer. Half-integrality based algorithms for cosegmentation of images. In *CVPR*, 2009. 2

[14] L. Mukherjee, V. Singh, and J. Peng. Scale invariant cosegmentation for image groups. In *CVPR*, 2011. 2

[15] C. Rother, V. Kolmogorov, T. Minka, and A. Blake. Cosegmentation of image pairs by histogram matching - incorporating a global constraint into MRFs. In *CVPR*, 2006. 1, 2

[16] S. Vicente, V. Kolmogorov, and C. Rother. Cosegmentation revisited: Models and optimization. In *ECCV*, 2010. 2

[17] S. Vicente, C. Rother, and V. Kolmogorov. Object cosegmentation. In *CVPR*, 2011. 2, 3, 6, 7

[18] S. N. P. Vitaladevuni and R. Basri. Co-clustering of image segments using convex optimization applied to em neuronal reconstruction. In *CVPR*, 2010. 2