Joint Cosegmentation and Cosketch by Unsupervised Learning

Jifeng Dai, Ying Nian Wu, Jie Zhou, and Song-Chun Zhu

Abstract—Cosegmentation refers to the problem of segmenting multiple images simultaneously by exploiting the similarities between the foreground and background regions in these images. The key issue in cosegmentation is to align the common objects in these images. To address this issue, we propose an unsupervised learning framework for cosegmentation, by coupling cosegmentation with what we call "cosketch." The goal of cosketch is to automatically discover a codebook of sketch templates shared by the input images. The sketch template is of hierarchical compositional structure where a large structured representational unit is composed of deformable smaller units. These sketch templates capture distinct image patterns and each template is matched to similar image patches in different images. The cosketches align foreground objects, thereby providing crucial information for cosegmentation. We present a statistical model whose energy function couples cosketch and cosegmentation. We then present an unsupervised learning algorithm that performs cosketch and cosegmentation by energy minimization. In experiments, we apply the proposed method to some public benchmarks on cosegmentation, including MSRC, iCoseg and ImageNet. We also test our method on a new dataset called Coseg-Rep where cosegmentation can be performed on a single image with repetitive patterns.

Index Terms—Cosegmentation, Cosketch, Unsupervised learning, Hierarchical Model, Sketch Model

1 Introduction

The goal of image segmentation is to partition the domain of an image into different regions with coherent visual patterns [1]. Image segmentation is intrinsically ambiguous without the guidance of high-level knowledge. Human intervention is usually required in various scenarios to provide additional information and reduce the ambiguities, for example, interactive segmentation [2], strongly supervised segmentation [3]. A new scenario studied recently is called cosegmnetation. Given a set of unlabeled images containing similar objects from the same object category, the goal of cosegmentation is to jointly segment the common objects in all the images. The repetition of common objects in these images provide mutual supervision for disambiguating the segmentations. Compared with interactive and supervised segmentation, cosegmentation requires very little human intervention and has attracted considerable attention from the vision community.

1.1 Motivation and Objectives

A key issue in cosegmentation is to find the "common" foreground objects by exploring the similarities in the visual patterns among the input images. In this paper, we present an unsupervised learning framework for cosegmentation. The main idea is to couple the task of cosegmentation with what we call *cosketch*. The goal of cosketch is to learn a codebook of sketch templates that are shared by the input images, and to sketch the images by these commonly shared templates. The sketch templates help establish correspondence between different images, providing crucial top-down information for segmentation.

To compute the co-sketch, we devide images into two categories of atomic elements as in the primal sketch representation [4]: i) sketches including region boundaries as well as non-boundary edges and lines; and ii) non-sketches including region interiors and shapeless patterns such as sky and water etc. We observe that objects, in existing datasets of cosegmentation, can be divided into three different categories according to their foreground complexity (see Fig. 1):

1) Objects with common sketch configurations, such as faces, leaves, pyramids etc.
2) Composite objects with common parts, such as street signs, houses etc. Although these objects do not have a common configuration, they share common parts.
3) Stochastic objects with common sketch elements, like trees, which do not have a common sketch configuration or parts but are stochastic with common sketch elements.

We propose to learn a codebook of hierarchical sketch templates to account for foreground objects of the three granularities above, which are then used to align the objects in different images. Fig. 2 illustrates our method. A codebook of two sketch templates (head and body) are learned from a set of input images of deer that are not a priori aligned or annotated, where each sketch template

- *Jifeng Dai is with the Department of Automation, Tsinghua University, Beijing, 100084. He was a visiting student to the Department of Statistics at University of California, Los Angeles when doing the majority part of this work. Email: daijifeng001@gmail.com*
- *Ying Nian Wu and Song-Chun Zhu are with the Department of Statistics, University of California, Los Angeles, Los Angeles, CA, 90095. E-mail: {ywu,sczhu}@stat.ucla.edu*
- *Jie Zhou is with the Department of Automation, Tsinghua University, Beijing, 100084. Email: jzhou@tsinghua.edu.cn*

(a)

(b)

(c)

Fig. 1: An example of input images for cosegmentation from different object categories in the MSRC dataset: (a) Faces, in which the common objects have a similar sketch configuration; (b) Signs, in which the foreground objects have different configurations of common parts; (c) Trees, in which the foreground objects are stochastic with common sketch elements.

is of a hierarchical compositional structure. These sketch templates capture distinct image patterns and the same template is matched to similar image patches in different images by translation, rotation, scaling and deformation. Each sketch template is associated with a segmentation template to be explained below, which provides top-down information for segmentation.

1.2 Overview of Our Method

Representation and model. We propose a statistical model that couples a sketch model and a region model. The model consists of the following three components.

(1) *Sketch model*. It seeks to encode the sketchable patterns of the input images by a codebook of sketch templates. Each sketch template is a large structured representational unit composed by a group of deformable smaller less-structured part templates, and each part template is composed of a group of shiftable atomic Gabor bases. It is a generative model with explicit variables for sketch deformations and is suitable for unsupervised learning.

(2) *Region model*. It represents the non-sketchable visual patterns. Each pixel of an input image is assigned a label indicating which region this pixel belongs to. The region model is defined conditional on the pixel label maps, which is in the form of a Markov random field that models the marginal distributions of pixel colors and pairwise similarities between neighboring pixels.

(3) *Coupling*. The sketch model and region model are coupled by associating each sketch template with a *segmentation template*. The coupling is not a hard assignment but probabilistic, which is in the form of a set of *probability maps* of pixel labels defined on the part templates of the sketch template. For each pixel within the bounding box of a part template, the probability map gives the probability that this pixel belongs to each region. The collection of the probability maps associated with a sketch template is called a segmentation template. Segmentation templates provide top-down prior information for pixel labels in the region model. Conversely, the pixel labels obtained from segmentation serve as data for the segmentation templates, and they provide bottom-up information for inferring sketch representation.

Unsupervised learning algorithm. Fitting the above model by energy minimization leads to a relaxation algorithm that alternates the following two steps.

(I) *Image parsing*: Given the current sketch templates, segmentation templates and the parameters for the sketch and region models, sketch the images by the sketch templates, and segment the images by graph cuts [5].

(II) *Re-learning*: Given the current image sketches and segmentations, re-learn the sketch templates, segmentation templates and model parameters.

The image parsing step itself consists of two sub-steps.

(I.1) *Sketch-guided segmentation*. Given the current sketches of the images by the sketch templates, segment the images by graph cuts with the associated segmentation templates as prior.

(I.2) *Segmentation-assisted sketch*. Given the current pixel labels of region segmentation, sketch the images by matching the sketch templates and the associated segmentation templates to the images and their label maps respectively.

Random initialization with no preprocessing. The sketch templates and the associated segmentation templates are initialized by learning from randomly cropped image patches, without any sophisticated pre-processing. Relaxation by energy minimization automatically results in alignment and segmentation, while distinct templates are being learned.

Experiments and data sets. We evaluate the proposed method on the **MSRC** [6], **iCoseg** [7] and **ImageNet** [8] datasets. Our method achieves state of the art accuracies on the MSRC and ImageNet datasets. To further test the proposed method, we collect a new dataset called **Coseg-Rep**, which contains 23 object categories with 572 images.

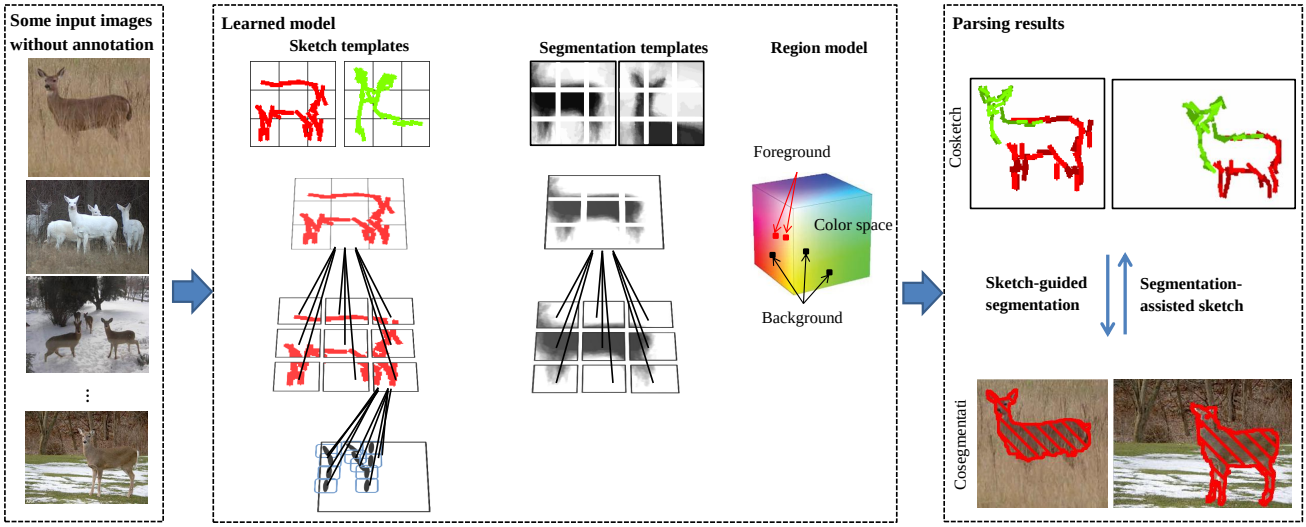


Fig. 2: An illustration of the proposed approach. Distinct sketch templates are learned (from 19 input images) and are matched to specific image patches in different images. Each sketch template is composed of 3×3 shiftable part templates, and each part template is composed of a set of shiftable Gabor bases. Sketch templates are coupled with segmentation templates that provide top-down clues for segmentation.

One special category contains 116 images such as tree leaves, where similar sketch patterns repeat themselves within the same image. As a result, cosegmentation can be performed on a single image. Our Coseg-Rep datasets, ground truth labels and results are available for the research community for further investigation[1].

1.3 Related Work and Comparisons

Our paper is related to the following four threads of research in the literature.

1, Shape models in strongly supervised segmentation. The advantage of utilizing shape model to provide top-down information for segmentation has been well demonstrated by a number of papers in strongly supervised segmentation, where the training images with ground truth annotations are given to train the generic segmentation model [3], [9]–[11] or to perform segmentation propagation [8]. In [3], [11], template-based models capturing high-level shape cues are trained from the aligned training images. However, it is hard to train such high level models that capture global information from non-annotated images. These methods do not work for the scenario of cosegmentation unlike our method.

2, Recent approaches in cosegmentation. Existing methods for cosegmentation mainly focus on employing local features. For example, in [12]–[20], where image features such as color histogram, SIFT, Fisher vectors etc. are extracted at all the pixels (or superpixels), so that those pixels (or superpixels) with similar features are encouraged to share the same segmentation results. One potential problem with the image features is that they may be too local to be distinctive, so they may not provide strong prior information for segmentation. In contrast, the explicit sketch templates used by our method cover much larger area (120×120 pixels) and capture much larger patterns, so that cosketch by these templates help to establish the correspondence between objects as well as their parts in different images. The associated segmentation templates thus provide strong prior information for cosegmentation. As a result, our approach achieves higher accuracies than existing state of the art methods on several challenging benchmarks.

There are some previous endeavors trying to learn high-level sketch templates from unannotated images to help segmentation. In [21], the edge model is defined by Gaussian distributions over Canny edge strength transformed by a deformation field. In [22], shape model is in the form of a rigid edge energy map covering regions determined by salient object detector. Both algorithms are only tested on images with roughly aligned object instances, which provide vital information for the learning of sketch templates. In contrast, our unsupervised learning method can be efficiently applied to non-aligned images where the object instances of the same category can be viewed from different perspectives, and can appear at different locations, orientations and scales in the input images. The correspondence between the images are automatically established while the sketch templates are being learned by the principled energy minimization algorithm, without any need for sophisticated initialization or preprocessing. In addition, our sketch model is based on a dictionary of hierarchical compositional templates, which is much more complex than those in [21], [22] and can better capture the variations of structure, viewpoint, deformation etc.

Very recently, Rubinstein et al. [23] utilized salient object detector and SIFT flow to provide high-level information for image cosegmentation. Salient object detector provides

1. http://www.stat.ucla.edu/~jifeng.dai/research/JointCosegmentationCosketch.html

foreground/background prior, and SIFT flow helps to establish correspondence between different images. Faktor and Irani [24] perform cosegmentation by composing an image by image parts pooled from a set of similar images. Compared to their methods, our approach is different in the following aspects. i) It explicitly models the sketchable visual patterns and their coupling with regions in a generative way. ii) It learns a common model over the input images instead of performing computationally expensive pairwise SIFT flow comparisons or image composition.

3, Learning sketchable patterns and repeated patterns in images. This work is related to learning sketchable patterns [25], [26] in images. In [25], [26], the sketchable visual patterns are represented by a collection of deformable Gabor bases, called active basis. Active basis models are learned from images that are roughly aligned.

This work is also related to [27]–[33], where repeated patterns are learned from unaligned input images. In [30]–[32], a set of HOG or active basis templates are learned from multiple input images of the same object category. In [33], recurring tuples of visual words are extracted from single image with repetitive patterns.

Different from our method, the above mentioned methods do not deal with the problem of segmentation. In addition, our sketch model is a codebook of hierarchical compositional templates, which is more flexible than those used in previous approaches.

4, Learning hierarchical compositional models. In this work, hierarchical compositional models are learned to represent the sketchable visual patterns. Hierarchical compositional models are very popular for modeling patterns of objects, see [34]–[40] for some examples. Many existing approaches to learning hierarchical compositional models are usually supervised where the object bounding boxes are given [39], [40] or weakly supervised where images are labeled and roughly aligned [34]. In this paper, we learn dictionaries of hierarchical compositional templates from unaligned natural images without annotations, which is more challenging.

Our work bears some similarities to [34], which seeks to organize the compositions of Gabor bases or edgelets into hierarchical structures. The hierarchical structures in [34] are learned layer-by-layer in a bottom-up manner. Once the lower layers are learned, they are fixed in the subsequent learning of higher layers. In our iterative learning algorithm, the part templets are re-learned and the Gabor bases are re-selected in each iteration, so the learning is more top-down than bottom-up. Please refer to Fig. 5 for the iterative learning process.

1.4 Contributions and Paper Organization

This paper makes the following contribution to the literature.

- It presents a statistical model whose energy function couples cosketch and cosegmentation. In cosketch, a codebook of generative sketch templates organized into hierarchical compositional structure is introduced to align common foreground objects.
- It presents an unsupervised learning algorithm that can efficiently perform cosketch and cosegmentation jointly on non-aligned images.
- In experiments, the proposed approach outperforms previous state of the art methods in cosegmentation on several challenging public benchmarks. It also create a new dataset named Coseg-Rep, with a special category that contains natural images with repetitive patterns. So we can test the algorithm for learning the common foreground model from a single image.

In comparison with a previous conference version [41], the paper expand the algorithm and the experiments in the following aspects: i) the sketch templates are extended to be of hierarchical compositional structure; ii) the proposed algorithm is tested on the challenging ImageNet dataset, and is improved in performance on the MSRC, iCoseg and Coseg-Rep datasets; and iii) it analyzes the cosegmentation accuracy with respect to several important influence factors.

The rest of this paper is organized as follows. Section 2 introduces the statistical model that couples cosketch and cosegmentation. Section 3 explains the unsupervised learning algorithm. Section 4 presents experiments on public benchmark datasets and introduces the Coseg-Rep dataset. Finally Section 5 concludes with a discussion.

2 Representation and Models

For clarity, this section and the next section present the simplest form of the model and algorithm. Implementation issues for the general situation will be treated at the end of Section 3.

2.1 Notation and Problem Definition

Let $\{\mathbf{I}^{(m)},\, m = 1, ..., M\}$ be a set of input images. Let Λ be image domain of $\mathbf{I}$, i.e., Λ collects all the pixels of $\mathbf{I}$. For each pixel $x \in \Lambda$, let $f(x)$ be the label of pixel x for image segmentation, $f(x) = 1$ if x belongs to foreground, and $f(x) = 0$ if x belongs to background. The task of cosegmentation is to return the label maps $\{f^{(m)}(x), m = 1, ..., M\}$ as output and is evaluated by $\{f^{(m)}\}$.

In the sketch model, $\mathbf{I}(x)$ is assumed to be a grey level intensity. In the region model, $\mathbf{I}(x)$ is assumed to be a three-dimensional vector in the color space.

2.2 Sketch Model

The sketch model generates the sketchable patterns by encoding an image $\mathbf{I}$ using a codebook of *sketch templates*, where each template is of hierarchical compositional structure (see Fig. 3). The sketch primitives at different layers of the model can be organized into layered dictionaries. Table 1 defines the primitives and elements, their parameters and the allowed ranges of values, which we shall elaborate in the following.

Layer ID	Template type	Parameters	Deformation Range	Template Size
$\Delta^{(5)}$, $\Delta^{(4)}$	Sketch template H	$l_k = \begin{pmatrix} \text{position } x_k \\ \text{scale } s_k \\ \text{orientation } o_k \end{pmatrix}$	$\Omega^{(4)} = \begin{pmatrix} \text{image domain } \Lambda \\ \times\{0.8, 1, 1.2\} \\ \times\{-2,-1,0,1,2\} \times [-\pi/16, \pi/16] \end{pmatrix}$	120×120 pixels
$\Delta^{(3)}$, $\Delta^{(2)}$	Part template A	$l_{t,v} = \begin{pmatrix} x_{t,v} \\ s_{t,v} \\ o_{t,v} \end{pmatrix}$ $\delta l_{k,v} = \begin{pmatrix} \delta x_{k,v} \\ 0 \\ \delta o_{k,v} \end{pmatrix}$	$\Omega^{(2)} = \begin{pmatrix} \{-2,0,2\} \times \{-2,0,2\} \text{ pixels} \\ \times\{-1,0,1\} \times [-\pi/16, \pi/16] \end{pmatrix}$	40×40 pixels
$\Delta^{(1)}$, $\Delta^{(0)}$	Gabor basis B	Basis coefficient $\lambda_{t,v,i}$ and normalizer $Z_{t,v,i}$ $l_{t,v,i} = \begin{pmatrix} x_{t,v,i} \\ s_{t,v,i} \\ o_{t,v,i} \end{pmatrix}$ $\delta l_{k,v,i} = \begin{pmatrix} \delta x_{k,v,i} \\ 0 \\ \delta o_{k,v,i} \end{pmatrix}$	$\Omega^{(0)} = \begin{pmatrix} \{-1,0,1\} \times \{-1,0,1\} \text{ pixels} \\ \times\{-1,0,1\} \times [-\pi/16, \pi/16] \end{pmatrix}$	13×13 pixels

TABLE 1: List of visual concepts used in our sketch templates, their parameters, deformation ranges and template sizes.

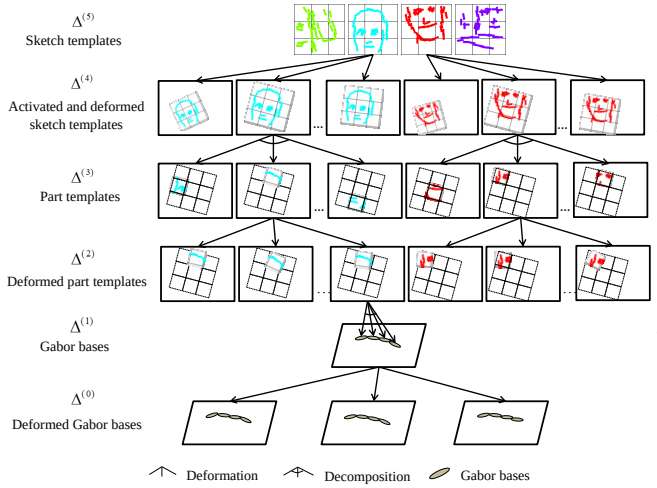


Fig. 3: Visual elements in layered dictionaries, which form a hierarchical compositional structure.

2.2.1 Layered dictionaries

$\Delta^{(5)}$ is the dictionary of sketch templates

$$\Delta^{(5)} = \{H_t, t = 1, ..., T\}, \tag{1}$$

in which t indexes the type, usually $T = 4$. The hierarchical templates capture the frequently occurring patterns in the input images.

$\Delta^{(4)}$ contains the spatially translated, rotated and scaled versions of the sketch templates in $\Delta^{(5)}$ for image representation. For an image $\mathbf{I}$, we encode it by K *activated sketch templates*, which are spatially translated, rotated and scaled copies of the sketch templates picked from $\Delta^{(5)}$. Let $H_{t_k}(l_k)$ be the k-th activated template of type t_k, where $l_k = (x_k, s_k, o_k)$ is its geometric attribute, where x_k is the location, s_k is the scale, and o_k is the orientation. Then the set $\partial H_t = \{H_t(l_k), l_k \in \Omega^{(4)}\}$ forms an equivalent class of H_t (i.e. the orbit w.r.t. a similarity transform group). $\Delta^{(4)}$ is the union of all possible activated templates:

$$\Delta^{(4)} = \cup_t \, \partial H_t, \quad \text{for} \quad H_t \in \Delta^{(5)}. \tag{2}$$

$\Delta^{(3)}$ denotes the dictionary of *part templates* of the activated sketch templates in $\Delta^{(4)}$. Let $dH_t \in \Delta^{(4)}$ be an activated (deformed) sketch template, and $A_{t,v}$ the v-th part template in dH_t. Then dH_t can be decomposed into

$$dH_t = (A_{t,1}(l_{t,1}), ..., A_{t,V}(l_{t,V})),$$

where V is the number of part templates, $l_{t,v} = (x_{t,v}, s_{t,v}, o_{t,v})$ is the geometric attribute of the v-th part template. $x_{t,v}$, $s_{t,v}$ and $o_{t,v}$ are the relative position, scale and orientation respectively. Here we fix the model structure by assigning 9 non-overlapping part templates arranged into a 3×3 grid to each sketch template. Then $\Delta^{(3)}$ is the collection of all the part templates

$$\Delta^{(3)} = \{A_{t,v}, t = 1, ..., T, v = 1, ..., V\}. \tag{3}$$

$\Delta^{(2)}$ includes all the shifted part templates. We allow each $A_{t,v}$ in $\Delta^{(3)}$ to translate and rotate within a small bounded range to account for object deformations in different images. Let $\delta l = (\delta x, 0, \delta o)$ be the shift within the bounded range, then we derive a set of shifted part templates $\{A_{t,v}(l_{t,v} + \delta l), \delta l \in \Omega^{(2)}\}$ for each $A_{t,v}$ in $\Delta^{(3)}$. Let $\partial A_{t,v}$ denote the equivalent class of $A_{t,v}$ subject to bounded shifts. Then $\Delta^{(2)}$ is the union of all the shifted part templates

$$\Delta^{(2)} = \cup_{t,v} \, \partial A_{t,v}, \quad \text{for} \quad A_{t,v} \in \Delta^{(3)}. \tag{4}$$

$\Delta^{(1)}$ contains the Gabor bases in the deformed part templates in $\Delta^{(2)}$. Following the active basis model in [25], the basis elements B are chosen to be Gabor bases at different positions and orientations. A deformed part template $dA_{t,v} \in \Delta^{(2)}$ is decomposed into a group of Gabor bases with zero mean and unit ℓ_2 norm

$$dA_{t,v} = (B(l_{t,v,1}), ..., B(l_{t,v,n})),$$

where $B(l_{t,v,i})$ is the i-th Gabor basis with $l_{t,v,i} = (x_{t,v,i}, s_{t,v,i}, o_{t,v,i})$. Here the $l_{t,v,i}$ for each $B(l_{t,v,i})$ is not pre-determined, but to be learned from the input images. Therefore, $\Delta^{(1)}$ is a set of Gabor basis elements decomposed from $\Delta^{(2)}$

$$\Delta^{(1)} = \{B(l_{t,v,i}) \in dA_{t,v}, \quad dA_{t,v} \in \Delta^{(2)}\}. \tag{5}$$

$\Delta^{(0)}$ contains the shifted Gabor bases in $\Delta^{(1)}$, which ground the templates onto image pixels. For each basis $B(l) \in \Delta^{(1)}$, we allow translations and rotations within bounded ranges and derive a shifted set $\partial B(l) = \{B(l + \delta l), \delta l \in \Omega^{(0)}\}$. Then $\Delta^{(0)}$ is the union of all these shifted Gabor bases

$$\Delta^{(0)} = \cup \partial B(l), \quad \text{for} \quad B(l) \in \Delta^{(1)}, \tag{6}$$

where each Gabor basis is the translated and rotated version of the original one.

As a summary, dictionaries $\Delta^{(j)}, j = 5, 4, 3, 2, 1, 0$ form a hierarchical compositional representation of the sketchable visual patterns. $\Delta^{(5)}$ is decomposed into $\Delta^{(3)}$, and $\Delta^{(3)}$ is decomposed to $\Delta^{(1)}$. $\Delta^{(4)}$ is the activated and shifted version of $\Delta^{(5)}$; while $\Delta^{(2)}$ and $\Delta^{(0)}$ are the shifted versions of $\Delta^{(3)}$ and $\Delta^{(1)}$ respectively.

2.2.2 Probabilistic modeling

Given an input image $\mathbf{I}$, we encode the sketches in $\mathbf{I}$ by K activated sketch templates. For now, let us assume that these K templates as well as their part templates and Gabor bases do not overlap with each other. The issue of overlapping will be considered later, which will not add anything conceptually.

Let Λ_k be the image domain covered by the k-th activated and deformed sketch template $H_{t_k}(l_k) \in \Delta^{(4)}$ to encode image $\mathbf{I}$. Then the image domain Λ of $\mathbf{I}$ can be divided into

$$\Lambda = \Lambda_0 \cup [\cup_{k=1}^{K} \Lambda_k], \tag{7}$$

where Λ_0 refers to the image domain not covered by any sketch templates.

Each activated template is further divided into part templates, which are also allowed to shift relative to each other to encode the input image. Let $\Lambda_{k,v}$ be the image domain covered by the shifted part template $A_{t_k,v}(dl_{k,v}) \in \Delta^{(2)}$, where $dl_{k,v} = l_k + l_{t_k,v} + \delta l_{k,v}$. Then the image domain Λ_k covered by $H_{t_k}(l_k)$ is divided into

$$\Lambda_k = \cup_{v=1}^{V} \Lambda_{k,v}. \tag{8}$$

Each shifted part template $A_{t_k,v}(dl_{k,v})$ is further divided into shiftable Gabor bases to ground onto image pixels. Let $\Lambda_{t,v,i}$ be the image domain covered by the shifted Gabor basis $B(dl_{k,v,i}) \in \Delta^{(0)}$, where $dl_{k,v,i} = dl_{k,v} + l_{t_k,v,i} + \delta l_{k,v,i}$. Then the image domain $\Lambda_{k,v}$ covered by $A_{t_k,v}(dl_{k,v})$ is divided into

$$\Lambda_{k,v} = \Lambda_{k,v,0} \cup [\cup_{i=1}^{n} \Lambda_{k,v,i}], \tag{9}$$

where $\Lambda_{k,v,0}$ refers to the empty pixels inside $\Lambda_{k,v}$ not occupied by the Gabor bases.

Let $\Lambda_S = \{\cup_{k,v,i} \Lambda_{k,v,i}\}$ denote the pixels covered by the Gabor bases in image $\mathbf{I}$, which correspond to the sketchable image areas, and let $\overline{\Lambda_S} = \{\Lambda_0 \cup [\cup_{k,v} \Lambda_{k,v,0}]\}$ denote the pixels not covered by the Gabor bases, which correspond to the non-sketchable image areas. The image is divided into two components

$$\mathbf{I} = (\mathbf{I}(\Lambda_S),\ \mathbf{I}(\overline{\Lambda_S})).$$

The activation and deformation states of the dictionaries at different layers form the sketch representation $W_S = (t_k, l_k, \delta l_{k,v}, \delta l_{k,v,i}, \forall k, v, i)$ of image $\mathbf{I}$. Here we define a probability model $p(\mathbf{I}|W_S)$ over W_S. Due to the tree structure of the hierarchical compositional model and the non-overlapping assumption, $p(\mathbf{I}|W_S)$ can be factorized as follows by assuming independence between the parts,

$$\begin{aligned} p(\mathbf{I}|W_S) &= p(\mathbf{I}(\overline{\Lambda_S}), \mathbf{I}(\Lambda_S)|W_S) \\ &= p(\mathbf{I}(\overline{\Lambda_S}))p(\mathbf{I}(\Lambda_S)|W_S) \\ &= p(\mathbf{I}(\overline{\Lambda_S})) \prod_{k,v,i} p\left(\mathbf{I}(\Lambda_{k,v,i})|B(dl_{k,v,i})\right). \end{aligned} \tag{10}$$

Following the active basis model [25], we take a reference model $q(\mathbf{I})$ for generic natural images, which can be factorized into the product of the patch probabilities $q\left(\mathbf{I}(\Lambda_{k,v,i})\right)$ as well as $q(\mathbf{I}(\overline{\Lambda_S}))$ under conditional independence assumption.

We compute the probability ratio

$$\frac{p(\mathbf{I}|W_S)}{q(\mathbf{I})} = \frac{\prod_{k,v,i} p\left(\mathbf{I}(\Lambda_{k,v,i})|B(dl_{k,v,i})\right)}{\prod_{k,v,i} q\left(\mathbf{I}(\Lambda_{k,v,i})\right)}. \tag{11}$$

Since $p(\mathbf{I}(\overline{\Lambda_S}))$ uses the same model as $q(\mathbf{I}(\overline{\Lambda_S}))$, it is canceled in the ratio.

As each image patch $\mathbf{I}(\Lambda_{k,v,i})$ is still high dimensional, we project it to a one dimensional probability ratio along the response of basis function $B(dl_{k,v,i})$

$$r_{k,v,i} = \left\|\langle \mathbf{I}(\Lambda_{k,v,i}), B(dl_{k,v,i})\rangle\right\|^2,$$

and the latter can be modeled by a one-dimensional exponential distribution following the information projection principle [25].

$$\begin{aligned} \frac{p\left(\mathbf{I}(\Lambda_{k,v,i})|B(dl_{k,v,i})\right)}{q\left(\mathbf{I}(\Lambda_{k,v,i})\right)} &= \frac{p(r_{k,v,i})}{q(r_{k,v,i})} \\ &= \frac{1}{Z_{t_k,v,i}} \exp\left\{\lambda_{t_k,v,i} h(r_{k,v,i})\right\}. \end{aligned} \tag{12}$$

The above model has four aspects.

- $q(r)$ is a histogram of filter responses pooled over a set of natural images. It has high probabilities near zero and has heavy tails.
- h is a sigmoid transform that saturates the large Gabor basis response to τ:

  $$h(x) = \tau \left[2/(1 + e^{-2x/\tau}) - 1\right].$$

  It has high response when the patch coincides with an edge/bar feature in the image.
- $\lambda_{t,v,i}$ reflects the importance of the corresponding Gabor basis element in the learned sketch template, and should be estimated by maximum likelihood so that the expectation $E_{\lambda_{t,v,i}}[h(r)]$ matches the corresponding observed mean response from covered image patches.
- $Z_{t,v,i}$ can be computed using numerical integration to normalize the one dimensional probability distribution $p(r) = q(r)\frac{1}{Z_{t,v,i}} \exp\{\lambda_{t,v,i} h(r)\}$.

Let $\Theta_S = (\lambda_{t,v,i}, Z_{t,v,i}, \Delta^{(j)}, \forall t, v, i, j)$ be the parameters for the sketch model, the log-likelihood ratio of image $\mathbf{I}$ encoded by W_S is

$$\ell(\mathbf{I}|W_S, \Theta_S) = \sum_{k=1}^{K} \ell(\mathbf{I} \mid H_{t_k}(l_k)), \tag{13}$$

in which

$$\ell(\mathbf{I} \mid H_{t_k}(l_k)) = \sum_{v=1}^{V} \sum_{i=1}^{n} \left[\lambda_{t_k,v,i} h(r_{k,v,i}) - \log Z_{t_k,v,i}\right]. \tag{14}$$

Energy function for sketch model. We define the energy function of the sketch model to be

$$E(\mathbf{I}|W_S, \Theta_S) = -\ell(\mathbf{I}|W_S, \Theta_S). \tag{15}$$

2.3 Region Model

The region model generates non-sketchable visual patterns, by modeling the marginal distributions of $\mathbf{I}(x)$ (here $\mathbf{I}(x)$ is a three-dimensional vector in the color space), and the pairwise similarities between neighboring pixels, conditioning on pixel labels for segmentation. The energy function of the region model is in the form of pair-potential Markov random field. It consists of two terms: the unary potential and the pairwise potential.

Unary potential. The unary potential models the marginal distribution of pixel colors conditional on the pixel labels by mixtures of Gaussian distributions. Let $g(v; \mu, \Sigma)$ denote a three-dimensional Gaussian density function with mean μ and variance-covariance matrix Σ, and ρ denote the prior of a Gaussian density function within the mixture model, the unary potential is

$$\begin{aligned} &\phi(\mathbf{I}(x)|f(x)) \\ &= -\log\left[\sum_{c=1}^{C} \rho_{f(x),c} g(\mathbf{I}(x); \mu_{f(x),c}, \Sigma_{f(x),c})\right], \end{aligned} \tag{16}$$

where $\theta_R = (\rho_{f,c}, \mu_{f,c}, \Sigma_{f,c})$ is an image specific color model. As a commonly used approximation, the sum operation in (16) can be replaced by max operation. The default value of C is set to be 5.

Pairwise potential. If pixels x and y are nearest neighbors (in this paper we use the nearest neighborhood) as denoted by $x \sim y$, then we want $\mathbf{I}(x)$ and $\mathbf{I}(y)$ to be different from each other if x and y belong to different regions. The pairwise potential is defined as

$$\begin{aligned} &\psi(\mathbf{I}(x), \mathbf{I}(y)|f(x), f(y)) \\ &= \mathbf{1}(f(x) \neq f(y)) \exp\left[-\frac{\|\mathbf{I}(x) - \mathbf{I}(y)\|_2^2}{2\sigma^2}\right]. \end{aligned} \tag{17}$$

where $\mathbf{1}()$ is the indicator function, $\|\cdot\|_2^2$ denotes the squared ℓ_2 distance between the colors of neighboring pixels, and σ^2 is taken to be the mean squared distance between neighboring pixels.

Energy function for region model. Define $W_R = (f(x), x \in \Lambda)$ to be the region representation of $\mathbf{I}$. Define $\Theta_R = (\theta_R)$ to be the parameters of the region model. The energy function for the region model is

$$\begin{aligned} E(\mathbf{I}|W_R, \Theta_R) = &\sum_{x} \phi(\mathbf{I}(x)|f(x)) \\ &+ \sum_{x \sim y} \psi(\mathbf{I}(x), \mathbf{I}(y)|f(x), f(y)). \end{aligned} \tag{18}$$

2.4 Coupling Sketch and Region Models

The generative model that involves both the sketch model and the region model can be written as: $p(W_S, W_R)p(\mathbf{I} \mid W_S, W_R)$. The prior model can be factorized into $p(W_S, W_R) = p(W_S)p(W_R|W_S)$, where $W_R = (f(x), x \in \Lambda)$ consists of pixel labels, and $W_S = (t_k, l_k, \delta l_{k,v}, \delta l_{k,v,i}, \forall k, v, i)$ consists of activation and deformation states of sketch templates. We couple them by modeling $p(W_R|W_S)$, where the templates provide prior information for pixel labels.

Segmentation templates as probability maps. For the codebook of the sketch template $\{H_t, t = 1, ..., T\}$, we associate a segmentation template with each H_t, where the segmentation template itself is of hierarchical compositional structure. Specifically, for the image domain of part template $A_{t,v}$ in H_t, the coupling model defines a probability map $P_{t,v}$ on it for each part template. $P_{t,v}(x, f) = \Pr(f(x) = f)$ gives the prior probability of the pixel label $f(x)$ for pixel x covered by $A_{t,v}$. The segmentation template of H_t is the collection of probability maps $\mathbf{P}_t = (P_{t,v}, v = 1, ..., V)$, which is also of hierarchical compositional structure.

W_S specifies the geometric transformation of the sketch templates. Then the same geometric transformation also transforms the corresponding segmentation templates.

Coupling energy function. Let $\Theta_C = (\mathbf{P}_t, t = 1, ..., T)$ be the segmentation templates, we define the coupling energy

$$E(W_R|W_S, \Theta_C) = -\sum_{k=1}^{K} \ell(f(x)|\mathbf{P}_{t_k}(l_k)). \tag{19}$$

in which

$$\ell(f(x)|\mathbf{P}_{t_k}(l_k)) = \sum_{v=1}^{V} \sum_{x \in \Lambda_{k,v}} \log P_{t_k,v}(x, f(x)). \tag{20}$$

Combined energy function. Let $W = (W_R, W_S)$, and let $\Theta = (\Theta_R, \Theta_S, \Theta_C)$. The combined energy function is:

$$\begin{aligned} E(\mathbf{I}, W|\Theta) = &\gamma E(\mathbf{I}|W_S, \Theta_S) + E(\mathbf{I}|W_R, \Theta_R) \\ &+ E(W_R|W_S, \Theta_C). \end{aligned} \tag{21}$$

Here we introduce a weighting parameter γ because the sketch model is a sparse model (default setting: the maximum number of basis functions in a sketch template is 60), whereas the region model and the coupling model are dense models defined on all the pixels (default setting: the number of pixels in $\mathbf{P}_t$ is 120×120). The parameter γ is introduced to balance these two terms (default value: $\gamma = 200$). One may consider that $E(\mathbf{I}, W|\Theta)$ defines

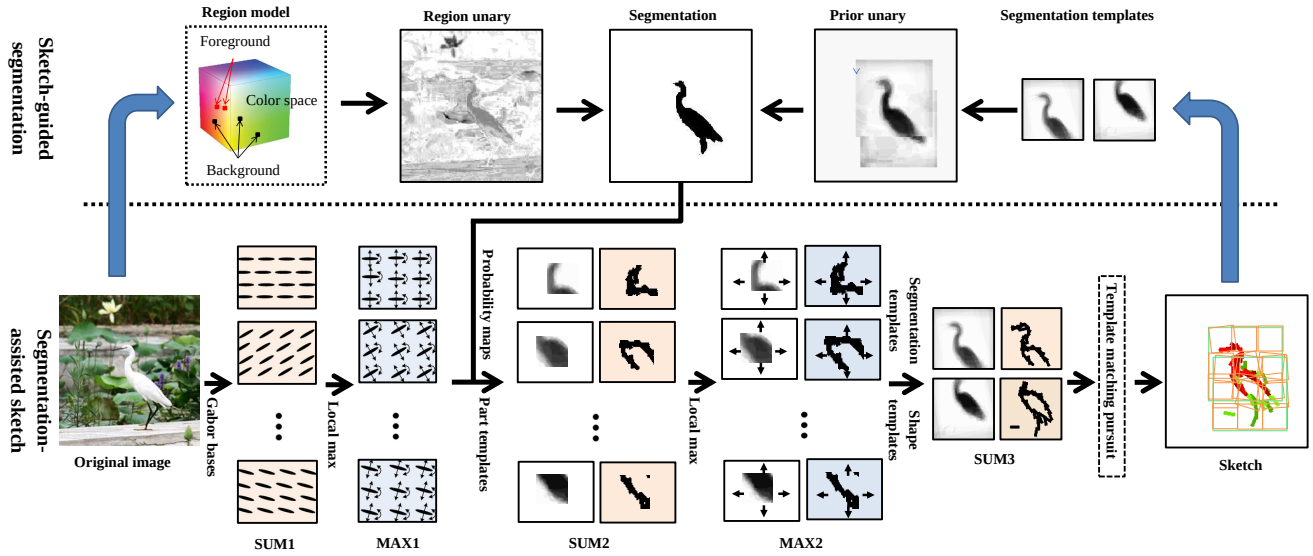


Fig. 4: Image parsing by sketch-guided segmentation and segmentation-assisted sketch. Sketch result helps to locate the foreground objects and provides top-down prior information for segmentation. Conversely, segmentation result provides bottom-up information for sketch.

a joint probability via the Gibbs distribution: $p(\mathbf{I}, W \mid \Theta) = \exp\{-E(\mathbf{I}, W|\Theta)/\gamma\}/Z(\Theta)$, where $Z(\Theta)$ is the normalizing constant.

3 Learning Algorithm

The input of the learning algorithm is $\{\mathbf{I}^{(m)}, m = 1, ..., M.\}$. The output includes $\{W^{(m)} = (W_S^{(m)}, W_R^{(m)}), m = 1, ..., M\}$ and $\Theta = (\Theta_S, \Theta_R, \Theta_C)$. The cosegmentation results are $\{W_R^{(m)}, m = 1, ..., M\}$.

The unsupervised learning algorithm seeks to minimize the total energy function $\sum_m E(\mathbf{I}^{(m)}, W^{(m)}|\Theta)$ over $\{W^{(m)}\}$ and Θ. The algorithm iterates the following two steps. (I) Image parsing: Given Θ, infer $W^{(m)}$ for each $\mathbf{I}^{(m)}$. (II) Re-learning: Given $\{W^{(m)}\}$, estimate Θ.

3.1 Image Parsing

The image parsing step is performed on each image $\mathbf{I}$, which can be further divided into two sub-steps. (I.1) Sketch-guided segmentation: Given W_S, infer W_R. (I.2) Segmentation-assisted sketch: Given W_R, infer W_S. An illustration of the image parsing algorithm is shown in Fig. 4. The issue of overlap between templates will be discussed at the end of this section.

I.1: Sketch-guided segmentation. This step minimizes

$$\begin{aligned} &E(\mathbf{I}|W_R, \Theta_R) + E(W_R \mid W_S, \Theta_C) \\ &= \left[\sum_x \phi(\mathbf{I}(x)|f(x)) - \sum_{k=1}^{K} \ell(f(x)|\mathbf{P}_{t_k}(l_k))\right] \\ &+ \sum_{x\sim y} \psi(\mathbf{I}(x), \mathbf{I}(y)|f(x), f(y)) \end{aligned} \tag{22}$$

over W_R. The energy function is in the form of a unary term and a pairwise term, which satisfies the submodular condition and can be efficiently optimized by graph cuts [5]. The sketch representation generates the prior distribution of pixel labels and adds to the unary term of the energy function of the region model $E(\mathbf{I}|W_R, \Theta_R)$. In this way, the vital alignment information provided by the sketch model is utilized to guide segmentation in a top-down manner.

I.2: Segmentation-assisted sketch. This step minimizes

$$\begin{aligned} &\gamma E(\mathbf{I}|W_S, \Theta_S) + E(W_R \mid W_S, \Theta_C) \\ &= -\sum_{k=1}^{K} \left[\gamma \ell(\mathbf{I} \mid H_{t_k}(l_k)) + \ell(f(x)|\mathbf{P}_{t_k}(l_k))\right] \end{aligned} \tag{23}$$

over W_S. The energy function is the weighted summation of sketch template and segmentation template matching scores. The segmentation template exploits pixel label information obtained from image segmentation via a bottom-up way to assist sketch.

We define the template matching score of sketch and segmentation template pair $(H_t, \mathbf{P}_t)$ over $\mathbf{I}$ to be:

$$\mathbf{R}_t(l) = \gamma \ell(\mathbf{I} \mid H_t(l)) + \ell(f(x)|\mathbf{P}_t(l)). \tag{24}$$

As the sketch template and segmentation template are both of tree structure, the template matching score can be calculated by a bottom-up template matching sub-process, which recursively performs SUM and MAX operations:

Procedure Template matching

Up-1 compute the Gabor basis matching score SUM1 of B on image $\mathbf{I}$:

$$\text{SUM1}(l) = |\langle \mathbf{I}, B(l) \rangle|^2.$$

Up-2 compute MAX1 by local maximization to account for local perturbations of Gabor bases:

$$\text{MAX1}(l) = \max_{\delta l} \text{SUM1}(l + \delta l).$$

Up-3 for $t = 1, ..., T, v = 1, ..., V$, compute matching scores $\text{SUM2}_{t,v}$ of the part template $A_{t,v}$ and the corresponding probability map $P_{t,v}$ on $\mathbf{I}$:

$$\text{SUM2}_{t,v}(l) = \gamma \sum_{i=1}^{n} [\lambda_{t,v,i} h(\text{MAX1}(l + l_{t,v,i})) - \log Z(\lambda_{t,v,i})] + \sum_{x \in \Lambda_{t,v}} \log P_{t,v}(x, f(x)).$$

Up-4 for $t = 1, ..., T, v = 1, ..., V$, compute $\text{MAX2}_{t,v}$ of $A_{t,v}$ and $P_{t,v}$ by local maximization to account for local perturbations of part templates:

$$\text{MAX2}_{t,v}(l) = \max_{\delta l} \text{SUM2}_{t,v}(l + \delta l).$$

Up-5 for $t = 1, ..., T$, compute the matching score SUM3_t of the sketch template H_t and the corresponding segmentation template $\mathbf{P}_t$ on $\mathbf{I}$:

$$\text{SUM3}_t(l) = \sum_{v=1}^{V} \text{MAX2}_{t,v}(l + l_{t,v}).$$

Set

$$\mathbf{R}_t(l) \leftarrow \text{SUM3}_t(l).$$

Suppose the k-th activation of sketch templates in the image $\mathbf{I}$ is known to be H_{t_k}, then the geometric attributes of H_{t_k} and its part templates $A_{t_k,v}$ can be determined by a top-down template localization sub-process of arg-max operations:

Procedure Template localization

Down-1 localize the sketch template H_{t_k} in image $\mathbf{I}$:

$$l_k = \arg\max_l \text{SUM3}_{t_k}(l).$$

Down-2 localize the part templates in image $\mathbf{I}$:

$$\delta l_{k,v} = \arg\max_{\delta l} \text{SUM2}_{t_k,v}(l + l_{t,v} + \delta l), \forall v.$$

Finally, a template matching pursuit algorithm is performed to sequentially select sketch templates to sketch $\mathbf{I}$.

Algorithm 1 Template matching pursuit

1: Initialize the maps of template matching scores $\mathbf{R}_t(l)$ for all (l, t) by the template matching sub-process. Let $k \leftarrow 1$.
2: Select the next best sketch template by finding the global maximum of the maps: $t_k = \arg\max_t [\max_l \mathbf{R}_t(l)]$.
3: Localize the selected template H_{t_k} in image $\mathbf{I}$ by the template localization sub-process and get $\{l_k, \delta l_{k,v} \forall v\}$.
4: Let the selected arg-max template inhibit overlapping candidate templates. If the candidate template $H_t(l)$ overlaps with $H_{t_k}(l_k)$, then set $\mathbf{R}_{(t)}(l) \leftarrow -\infty$.
5: Stop if all $\mathbf{R}_t(l) \leq 0$. Otherwise let $k \leftarrow k + 1$, and go to Step 2.

When performing cosegmentation on multiple images, we require that each template in the codebook can only be used once for each image. So if a template $H_{t_k}(l_k)$ is selected, we set $\mathbf{R}_{t_k}(l) \leftarrow -\infty$ for all l.

3.2 Re-learning

This step seeks to minimize the total energy function $\sum_m E(\mathbf{I}^{(m)}, W^{(m)}|\Theta)$ over Θ_S, Θ_R and Θ_C given $\{W_S^{(m)}\}$ and $\{W_R^{(m)}\}$. These three parameters are decoupled so the minimizations can be carried out separately.

II.1: Re-learn sketch templates. For each $t = 1, ..., T$, we re-learn H_t from all the image patches that are currently covered by H_t. Specifically, we re-learn each part template $A_{t,v}$ in H_t from the aligned image patches by the shared matching pursuit algorithm [25]. Let $\{\widehat{\mathbf{I}}_{u,t,v}, u = 1, ..., U\}$ denote the aligned image patches cropped from $\{I^{(m)}\}$ covered by $A_{t,v}$, the shared matching pursuit algorithm sequentially selects the Gabor bases and estimates the associated parameters. Each iteration seeks the maximal increase of the total log-likelihood. The algorithm is as follows.

Algorithm 2 Shared matching pursuit

1: Initialize $i \leftarrow 0$. For $u = 1, ..., U$, initialize the response maps $\widehat{R}_{u,t,v}(l) \leftarrow |\langle \widehat{\mathbf{I}}_{u,t,v}, B(l) \rangle|^2$ for all l.
2: $i \leftarrow i + 1$. Select the next basis function by finding

$$l_{t,v,i} = \arg\max_l \sum_{u=1}^{U} \max_{\delta l} h(\widehat{R}_{u,t,v}(l + \delta l)),$$

where $\max_{\delta l}$ is local maximum pooling within the bounded range of perturbations.
3: For $u = 1, ..., U$, given $l_{t,v,i}$, infer the perturbations by retrieving the arg-max in the local maximum pooling of Step 2:

$$\delta l_{u,t,v,i} = \arg\max_{\delta l} \widehat{R}_{u,t,v}(l_{t,v,i} + \delta l).$$

Let $dl_{u,t,v,i} = l_{t,v,i} + \delta l_{u,t,v,i}$ and the response $r_{u,t,v,i} \leftarrow \widehat{R}_{u,t,v}(dl_{u,t,v,i})$. Then let the arg-max basis inhibit nearby basis by setting $\widehat{R}_{u,t,v}(l) \leftarrow 0$ if the correlation $|\langle B(l), B(dl_{u,t,v,i}) \rangle|^2 > \epsilon$ (default: $\epsilon = .1$ to enforce the approximate orthogonality of Gabor bases).
4: Compute $\lambda_{t,v,i}$ by solving the maximum likelihood equation $E_{\lambda_{t,v,i}}[h(r)] = \sum_{u=1}^{U} h(r_{u,t,v,i})/U$. And derive the corresponding $Z_{t,v,i}$ by solving $p(r) = q(r) \frac{1}{Z_{t,v,i}} \exp\{\lambda_{t,v,i} h(r)\}$.
5: Stop if $\lambda_{t,v,i} [\sum_{u=1}^{U} h(r_{u,t,v,i})/U] - \log Z_{t,v,i} \leq 0$, else go back to Step 2.

II.2: Re-learn marginal distributions of regions. For foreground and background, fit the corresponding mixture of Gaussian distributions using the EM algorithm.

II.3: Re-learn segmentation templates. The probability map $P_{t,v}$ associated with each $A_{t,v}$ is learned from the pixel labels of all the image patches explained by $A_{t,v}$. Let $\{\widehat{f}_{u,t,v}(x), u = 1, ..., U\}$ denote the aligned label map patches cropped from $\{f^{(m)}(x)\}$ covered by $P_{t,v}$, $P_{t,v}$ is estimated by:

$$P_{t,v}(x, f) = \frac{1}{U} \sum_{u=1}^{U} \mathbf{1}(\widehat{f}_{u,t,v}(x) = f). \quad (25)$$

Initialization. For Θ_S, H_t and the associated parameters are learned from randomly cropped image patches. For Θ_R, the marginal distributions of foreground and background can either be initialized from a weak prior or a strong prior.

By default, we start from a weak prior, which learns the background distribution from pixels within 10 pixels from image boundaries and the foreground distribution from pixels covered by the aforementioned random patches. For strong prior, we initialize the marginal distributions from segmentation results in [23] on the MSRC and iCoseg datasets. The label maps are then initialized by graph cuts. For Θ_C, $\mathbf{P}_t$ is learned from the label maps of the aforementioned random patches.

Since the learning algorithm starts from a dictionary of templates learned from randomly cropped image patches, the initial templates are random and meaningless, and the differences among them are small. However, as the algorithm proceeds, the small differences among the initial templates quickly start a polarizing or specializing process, where the templates become more and more different, and they specialize in encoding different image elements.

An example of the iterative procedure of the proposed approach is shown in Fig. 5.

3.3 Implementation Issues

Overlap. In the template matching pursuit algorithm, after a candidate template $H_{t_k}(l_k)$ is selected, instead of letting $H_{t_k}(l_k)$ inhibit all overlapping candidate templates $H_t(l)$, we only let $H_{t_k}(l_k)$ inhibit nearby candidate templates. Specifically, we set all $\mathbf{R}_t(l) = -\infty$ if $\|x - x_k\|_2 < \rho D$, where $\|\cdot\|_2$ is the Euclidean distance, D is the side length of the sketch template, and ρD is the pre-set overlapping distance (default setting: $\rho = .4$).

In sketch guided segmentation, when generating the prior probabilities of pixel labels, it is possible that a pixel is covered by multiple overlapping part templates from one or multiple sketch templates. In that case, we first choose the sketch template with the highest log-likelihood (template matching score). Then the probability map associated with the highest responded part template is chosen to generate the prior probability for this pixel.

Resolution and rotation. We scan the sketch template H_t over $\mathbf{I}$, as well as the associated segmentation template $\mathbf{P}_t$ over the label map $f(x)$. We do it on multiple resolutions of $\mathbf{I}$ (default setting: we first resize each image to approximately 170^2 pixels, and then use three resolutions, which are .8, 1, 1.2 relative to the resized image). In each step of the template matching pursuit algorithm, we also maximize over the resolutions, and we place the selected template at the optimal resolution. After that, we map the sketch and segmentation templates back to the highest resolution, and perform inhibition and image segmentation at this resolution.

In addition to resolution, we also allow the templates to rotate (default range: $\{-2, -1, 0, 1, 2\} \times [-\pi/16, \pi/16]$).

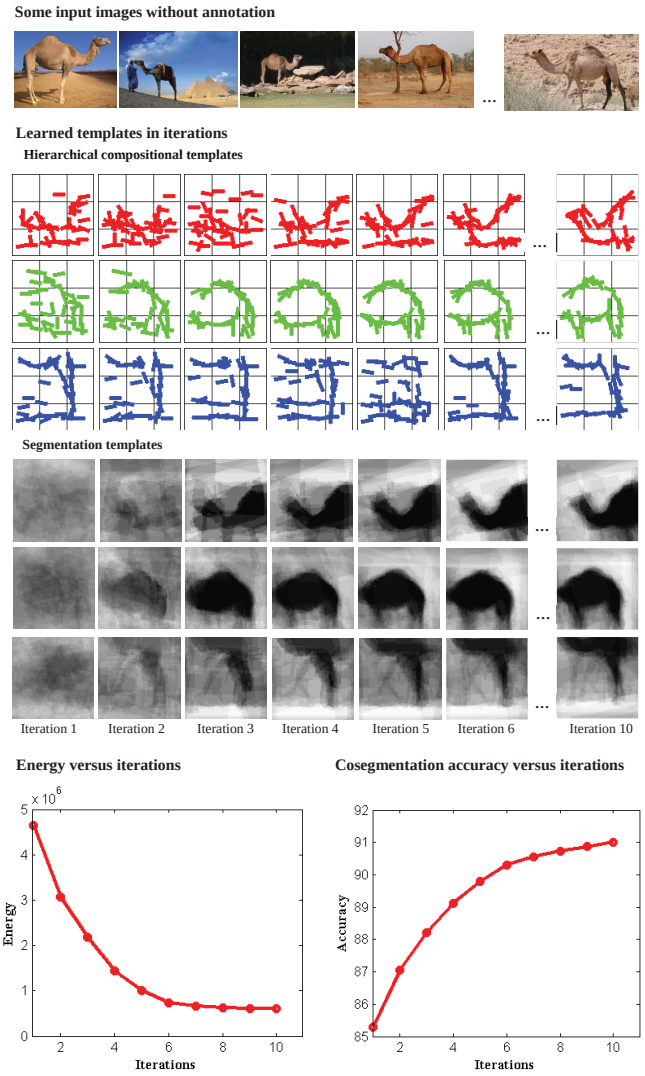


Fig. 5: Iterative procedure of the proposed approach on 24 input images of camel. As the algorithm iterates, the learned sketch templates and segmentation templates become more and more meaningful, the overall energy decreases, and the cosegmentation accuracy increases.

4 Experiments

4.1 Cosegmentation on MSRC and iCoseg

The MSRC dataset has 14 object classes with about 30 images per class, and was first introduced in the context of strongly supervised segmentation [6]. The iCoseg dataset has 30 classes with varying number of images per class, and was initially introduced for interactive cosegmentation [7]. In both datasets, instances are of varying appearances, locations, deformations and in cluttered backgrounds. There have been different evaluation protocols employed by different cosegmentation algorithms. Here for clarity and fair comparison, we compare with the unsupervised cosegmentaion algorithms without interactive input or additional annotated training images, and follow the evaluation protocol employed by the very recent approach in [23]: i) All the images in the above mentioned classes of MSRC and iCoseg are utilized for evaluation; ii) Segmentation accuracy is measured by the ratio of correctly labeled pixels of foreground and background with respect to the total number of pixels.

TABLE 2: Correctly labeled pixel ratios of the proposed approach, its stripped down versions, and the methods in [17], [20], [23], [24] on the MSRC dataset.

Class	Ours	Region only	Region+Rigid sketch [41]	[20]	[17]	[23]	[24]
Bike	77.6	77.8	76.7	68.3	38.0	**78.0**	76.9
Bird	**95.1**	92.2	94.0	74.4	28.0	93.5	92.7
Car	**94.2**	85.8	91.1	79.4	53.4	83.7	85.2
Cat	86.7	89.2	86.4	74.6	39.5	**90.4**	90.0
Chair	**91.0**	85.5	90.4	68.2	69.0	87.6	90.9
Cow	92.4	92.1	92.0	83.2	79.1	94.1	**96.7**
Dog	**93.0**	90.2	92.1	75.7	37.6	90.0	89.7
Face	**92.5**	86.7	89.0	83.5	58.3	82.0	88.6
Flower	**89.7**	85.8	88.9	65.9	36.0	85.5	88.6
House	88.7	86.6	87.0	58.4	56.7	87.2	**93.5**
Plane	**86.8**	86.2	86.4	52.9	56.1	86.6	85.5
Sheep	94.1	92.0	94.3	89.7	86.1	92.4	**95.1**
Sign	92.9	91.6	92.1	74.9	59.6	92.8	**94.1**
Tree	84.4	83.4	83.9	81.3	67.8	83.4	**85.9**
Average	**89.9**	87.5	88.9	73.6	54.6	87.6	89.2

TABLE 3: Correctly labeled pixel ratios of the proposed approach, its stripped down versions, and the methods in [17], [20], [23] on the iCoseg dataset. [41] is the Conference version of the proposed method.

	Ours	Region only	Region+Rigid sketch [41]	[20]	[17]	[23]
Average	**91.0**	88.9	90.1	70.2	70.4	89.8

In Table 2 and Table 3, we compare several stripped down versions of our model: (1) region only: using only the region model; (2) region+rigid sketch: using both the region and a rigid sketch model, where the sketch model is a dictionary of rigid sketch templates, which is the same as [41]. Using only the region model achieves average accuracies of 87.5% and 88.9% on the MSRC and iCoseg datasets respectively. By coupling with rigid sketch model, the accuracies are improved to 88.9% and 90.1%. The full model coupling both the region and the hierarchical compositional sketch model achieves average accuracies of 89.9% and 91.0% on the MSRC and iCoseg datasets respectively.

Table 2 and Table 3 also report the segmentation accuracies of recent state of the art works [17], [20], [23], [24]. The results of [17], [20], [23] are taken from [23]. The results of [24] are taken from [24]. On MSRC, our proposed approach surpasses the other methods in 7 out of 14 categories. And its average accuracy is 16.3%, 35.3%, 2.3% and 0.7% higher than the methods in [20], [17], [23] and [24] respectively. On iCoseg, the average accuracy of the proposed approach is 20.8%, 20.6% and 1.2% higher than the methods in [20], [17] and [23] respectively[2]. Note that the comparing methods bring in rather heuristic energy terms and computation procedures to boost the performance. While the proposed model is motivated from a generative perspective and is learned by the principled energy minimization algorithm.

Fig. 6 show some cosketch and cosegmentation results of the proposed approach on the images shown in Fig. 1. It can be seen that our proposed approach can effectively perform cosegmentation and cosketch despite that the object instances in the images are of varying appearances, locations, deformations and in cluttered backgrounds.

2. Following a data split protocol different from that in [23] and in this paper, [24] reports an average accuracy of 92.8% on iCoseg.

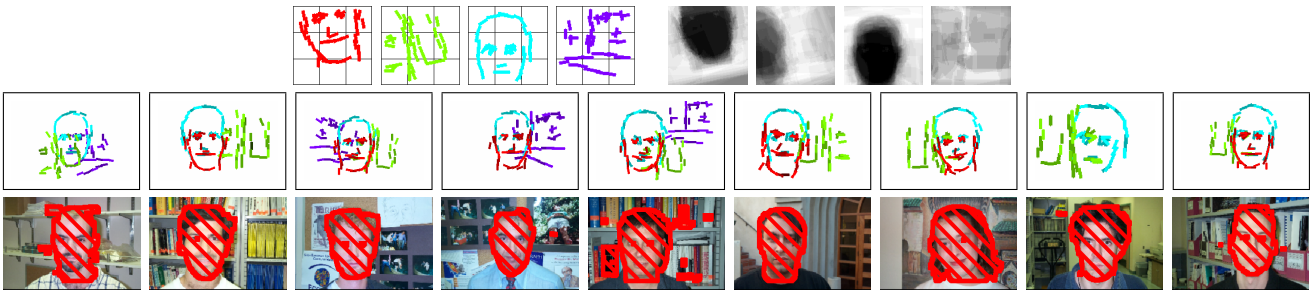
4.2 Cosegmentation on ImageNet

ImageNet is a challenging large-scale dataset of object classes, which contains millions of images annotated by the class label of the main object. The original ImageNet dataset does not have ground-truth annotations of segmentation. In [8], a subset of ImageNet is labeled with ground-truth segmentations. The test set contains 10 random images from each of 446 classes, for a total of $4,460$ images. It is very challenging due to limited training examples per class, huge visual variability and cluttered backgrounds. In [8], 60k images annotated with bounding boxes, 440k images with class labels and the semantic structure of class labels in ImageNet are utilized to provide strong supervision for segmentation propagation.

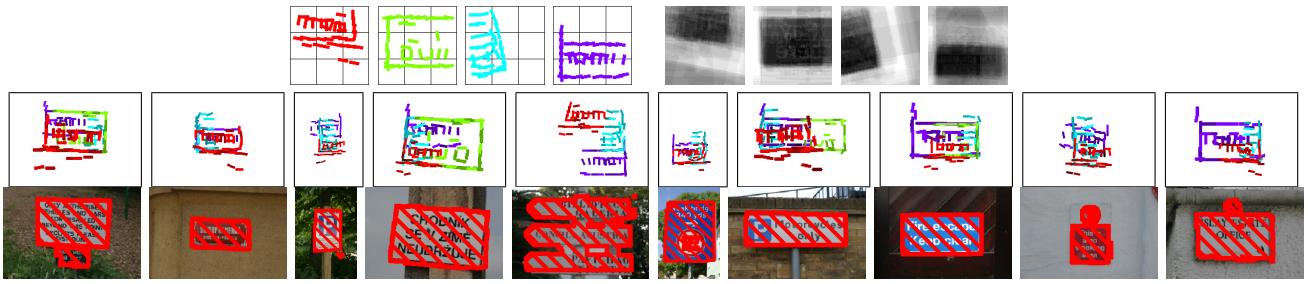
Here, we perform cosegmentation on the full test set without any additional supervision. Segmentation accuracy is measured by the correctly labeled pixel ratio, following the criterion in [8]. The average accuracies of the proposed approach, the supervised segmentation propagation algorithm in [8] and the Grabcut [2] baseline are presented in Table 4, where the results of segmentation propagation and Grabcut are taken from [8]. Our approach delivers the state of the art average accuracy of 79.0%, which is 1.9% and 8.0% higher than the supervised segmentation propagation algorithm in [8] and the Grabcut baseline respectively. Note that the previous state of the art result in [8] is achieved with the help of abundant supervision information, while our approach outperforms it *without any supervision*. Some parsing results of the proposed approach on ImageNet dataset are shown in Fig. 7.

4.3 Cosegmentation on Coseg-Rep

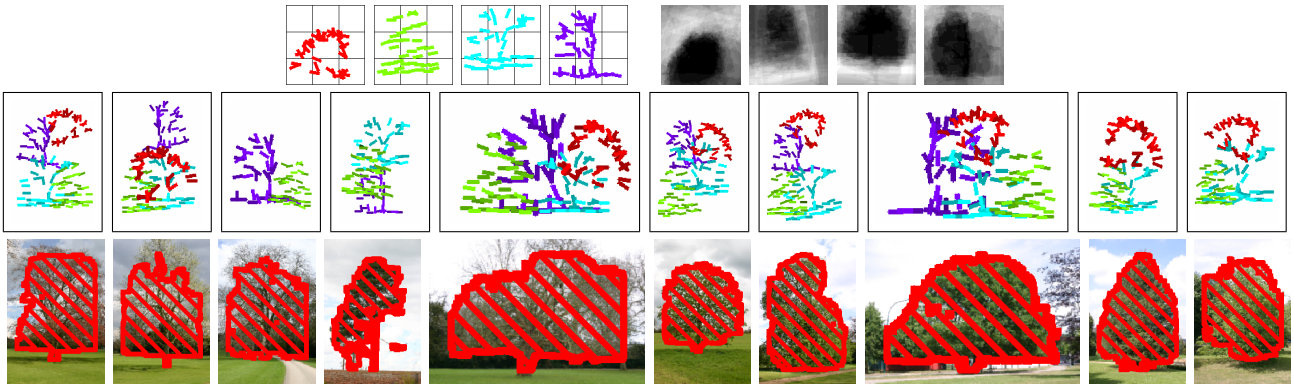
To further test our method, we collected a new dataset called Coseg-Rep, which has 23 object categories with 572 images. Among them, 22 categories are different species of animals and flowers, and each category has 9 to 49 images.



(a)



(b)



(c)

Fig. 6: Learned templates, cosegmentation and cosketch results of the proposed approach on the images shown in Fig. 1: (a) Faces; (b) Signs; (c) Trees.

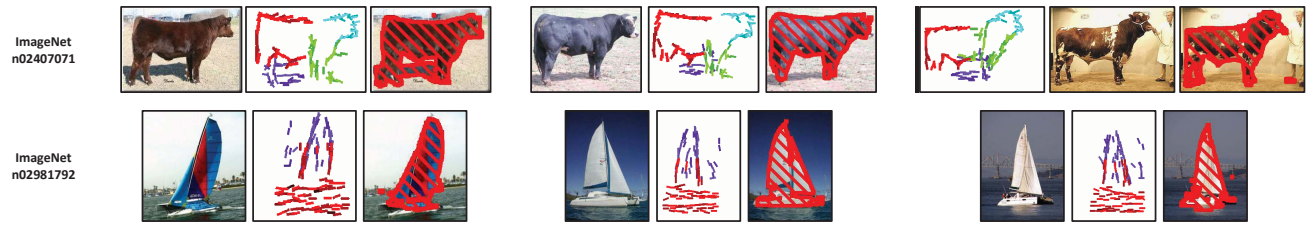


Fig. 7: Some cosketch and cosegmentation examples in the ImageNet dataset.

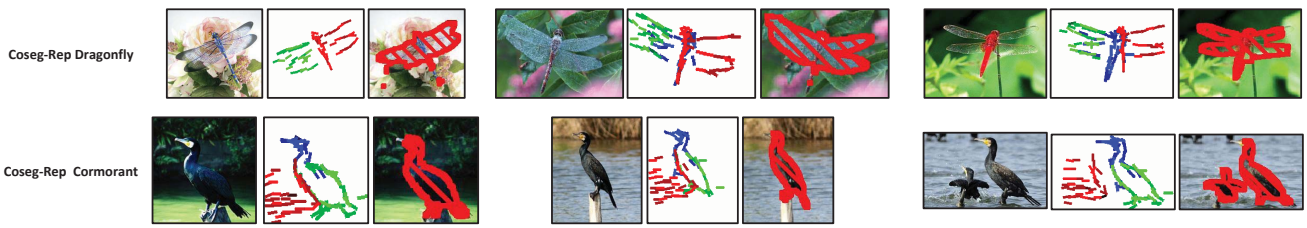


Fig. 8: Some cosketch and cosegmentation examples in the Coseg-Rep dataset.

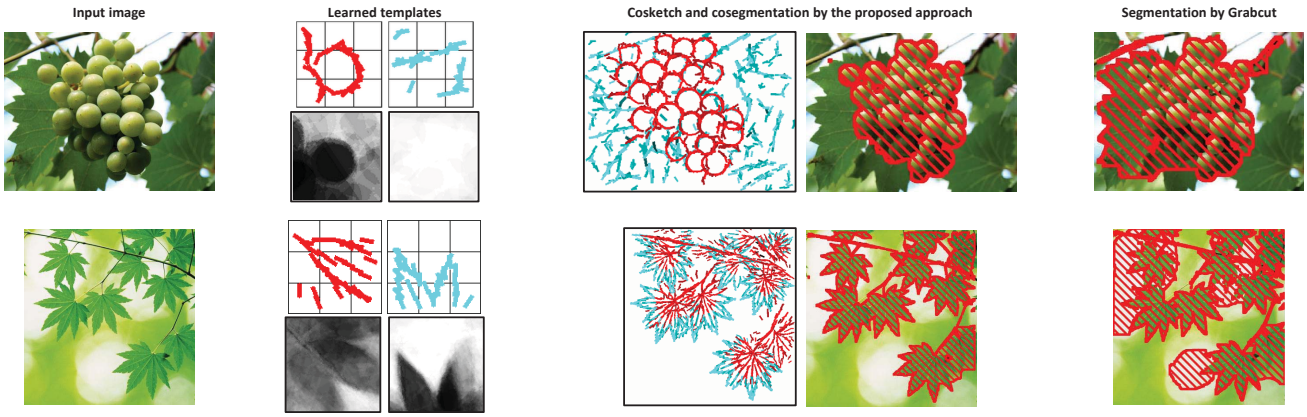
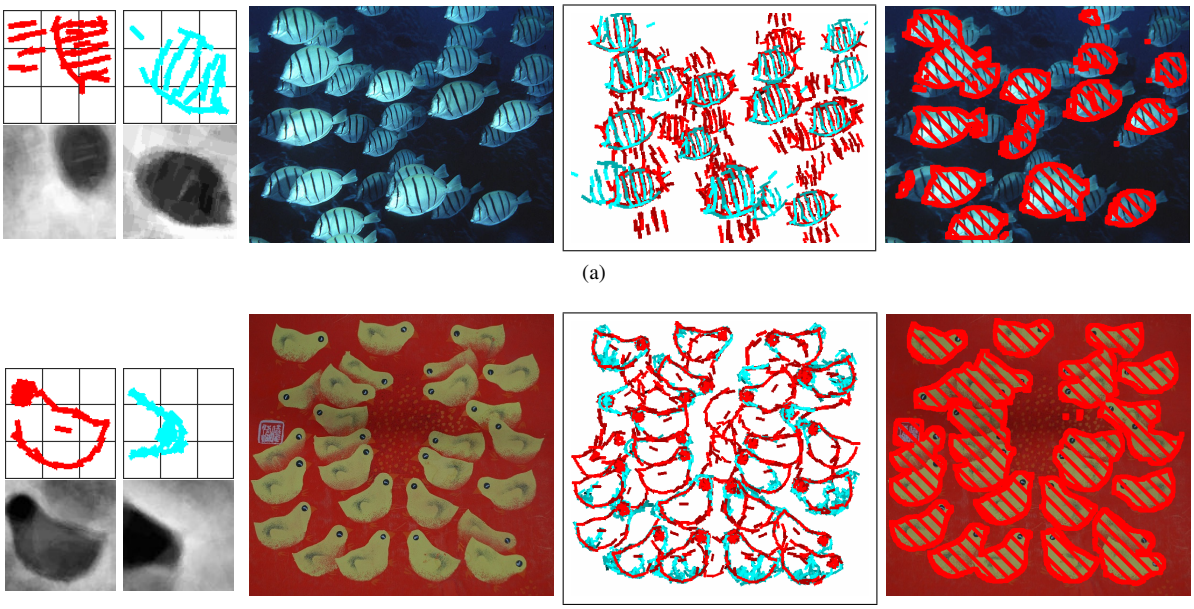


Fig. 9: Learned templates and corresponding parsing results of the proposed approach on two images of the repetitive category in the Coseg-Rep dataset. More accurate segmentation is achieved than the Grabcut [2] baseline.



(a)

(b)

Fig. 10: Learned templates, cosegmentation and cosketch results of the proposed approach on images with repetitive patterns used in [27] (shown in (a)), and [33] (shown in (b)).

TABLE 4: Correctly labeled pixel ratios of the proposed approach, the supervised segmentation propagation algorithm in [8] and the Grabcut [2] baseline on the ImageNet dataset. The algorithm in [8] utilizes an additional annotated dataset and the semantic structure of class labels in ImageNet for training, while our method does not use any supervision.

	Ours	[8]	[2]
Average	**79.0**	77.1	71.0

More important, there is a special category called "repetitive," which contains 116 natural images where similar sketch patterns repeat themselves within the same image instead of across different images, such as tree leaves and grapes etc. Segmentation of a single image with repetitive patterns is an important step for applications like automatic leaves recognition [42], editing of repetitive objects in a single image [43], latent print segmentation [44] and etc. Cosegmentation results of our proposed approach are presented in Table 5 and some examples are shown in Fig. 8. The mean accuracy over all the 23 categories is 91.7%, which is evaluated by the correctly labeled pixel ratio. Fig. 9 shows the learned templates and the corresponding parsing results on two images with repetitive patterns. Meaningful templates and satisfactory parsing results can be obtained although the algorithm starts from random initialization. As a comparison, our method gives more accurate segmentation result than a Grabcut [2] baseline method where the bounding box is set to be 10 pixels away from the boundary.

We also tested on the images collected in [27], [33]. The algorithms in [27], [33] seek to learn repetitive patterns from a single image and do not deal with segmentation. The learned templates and image parsing results of the proposed

TABLE 5: Correctly labeled pixel ratios of the proposed approach on the Coseg-Rep dataset.

Class	**Repetitive**	Blueflagiris	Camel	Cormorant	Cranesbill	Deer	Desertrose	Dragonfly	Egret	Firepink	Fleabane	Forgetmenot	Frog	Geranium	Ostrich	Pearblossom	Piegon	Seagull	Seastar	Silenecolorata	Snowowl	Whitecampion	Wildbeast	Average
Images	116	10	24	14	18	19	49	14	20	15	19	47	20	33	22	23	19	14	9	15	20	18	14	
Accuracy	89.2	97.2	91.5	89.9	96.1	85.3	96.4	85.9	93.9	98.9	95.9	95.6	86.2	98.6	93.1	92.3	83.0	89.3	92.3	97.0	71.2	93.7	96.8	91.7

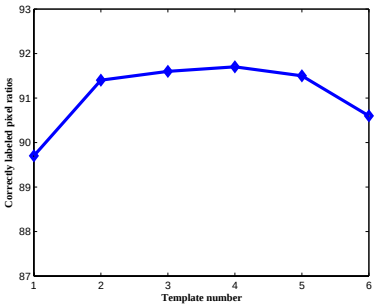


Fig. 11: Template number versus cosegmentation accuracy on the Coseg-Rep dataset.

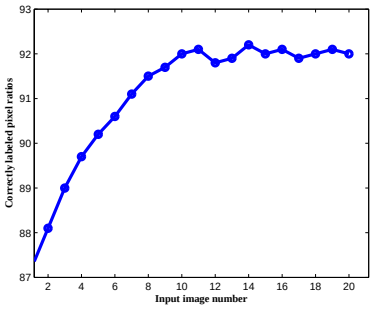


Fig. 13: Input image number versus cosegmentation accuracy on a subset of the Coseg-Rep dataset.

approach are shown in Fig. 10.

4.4 Parameter Settings and Limitations

Choice of template number T. Template number T controls the complexity of the sketch model and the coupling model. It is natural to ask how many templates are required to capture the sketchable visual patterns so as to get satisfactory cosegmentation accuracy. In general, it depends on how diverse the input images are and how many training examples we have, and in principle, the question of choosing T can be answered by the Bayesian information criterion. Here we adopt an experimental approach by analyzing the cosegmentation accuracy versus the number of templates on the Coseg-Rep dataset, where different classes have varying image numbers and image diversity. The experimental result is shown in Fig. 11. It can be seen that the cosegmentation accuracy is relatively robust with respect to different template numbers. A specific example of templates learned for varying template numbers is shown in Fig. 12. In general, we found that on the current public cosegmentation datasets, choosing template number T to be $2 \sim 4$ delivers satisfactory performance.

Robustness with respect to image number M. Since the goal of cosegmentation is to segment multiple images simultaneously by exploiting the similarities between these images, an interesting question is how many input images are needed to reliably learn the model. To our best knowledge, there is no answer to this question in the cosegmentation field. Here we study this question on the proposed approach by performing the following experiment: the 9 object categories with no less than 20 images except for "repetitive" category in the Coseg-Rep dataset are picked out. Then M images (M varies from 1 to 20) are independently drawn from each category to perform cosegmentation. We do this for 10 repetitions and the curve of mean cosegmentation accuracy over all the categories in all the rounds versus image number M is shown in Fig. 13. It can be seen that the mean cosegmentation accuracy generally improves as more and more input images are available. When $M = 1$, the problem of cosegmentation degrades to single image segmentation and the performance is not good; when M increase from 1 to 11, the mean cosegmentation accuracy improves steadily; when M is more than 11, the model can be estimated well and the benefit brought by more input images saturates.

Robustness with respect to image diversity. Another important question for cosegmentation is the robustness of the cosegmentation algorithm with respect to diversity in the input images. The input images of the same object category might well be quite visually different due to different visual subcategories, different viewing perspectives, deformations etc. We argue that a good cosegmentation algorithm should have strong robustness with respect to image diversity so as to be practical in real applications.

For our model, since the learned templates are usually parts of the objects instead of the whole objects, they can be shared in different poses. Different poses can also be captured by different templates in the codebook. The hierarchical compositional structure can deal with deformations by perturbing the parts and Gabor bases. In addition, the segmentation templates is automatically learned to be adaptive to the quality of the sketch templates. When the sketch templates are of good quality, the learned segmentation templates are assertive so the influence of sketch on segmentation is strong and vice versa.

Some failure examples. Fig. 14 shows some failure examples of our method. In these examples, the cosketch results are incorrect due to cluttered backgrounds and occlusions, which leads to false priors for cosegmentation.

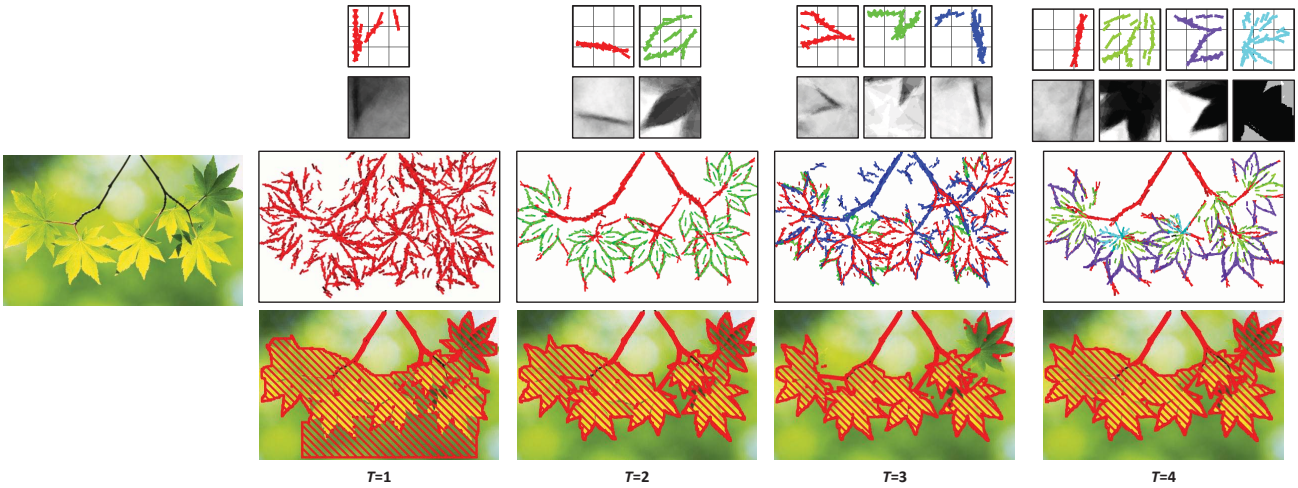


Fig. 12: The learned templates and the corresponding parsing results of the proposed approach for varying numbers of templates ($T = 1 \sim 4$).

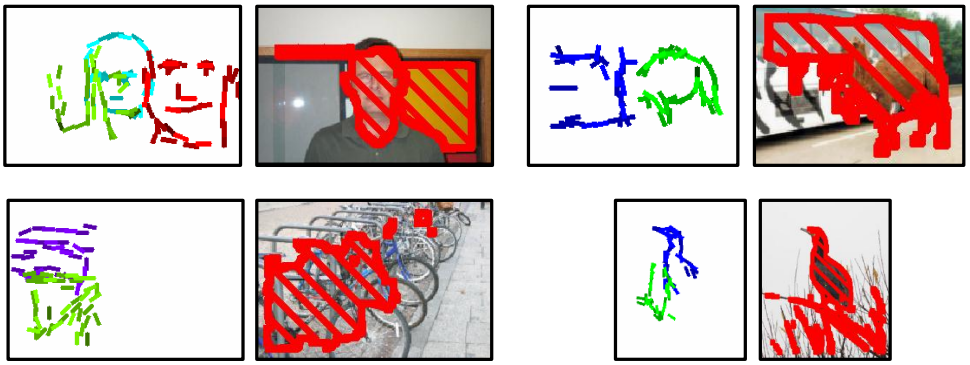


Fig. 14: Some failure examples of the proposed approach. The cosketch results are incorrect due to cluttered background and occlusions, which cause false priors for cosegmentation.

5 Conclusion

In this paper, we present a statistical model whose energy function couples cosketch and cosegmentation. In cosketch, sketch templates of hierarchical compositional structure are unsupervisedly learned from the input images. The sketch templates are coupled with region models to provide top-down information for cosegmentation. In experiments, we demonstrate that the proposed approach can efficiently perform cosegmentation on challenging datasets and achieve state of the art accuracy on several public benchmarks.

Currently, we utilize a one layer location invariant region model for the non-sketchable visual patterns. In the future, we plan to extend the region model to a location sensitive hierarchical compositional model as well.

Acknowledgments. The authors thank for the research grants: NSF DMS 1310391, ONR MURI N00014-10-1-0933, DARPA MSEE FA8650-11-1-7149, NSFC 61225008, NSFC 61020106004, MOEC 20120002110033 and China Scholarship Council.

References

[1] Z. Tu and S.-C. Zhu, "Image segmentation by data-driven markov chain monte carlo," *PAMI*, vol. 24, no. 5, pp. 657–673, 2002.

[2] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," in *TOG*, 2004.

[3] M. Kumar, P. H. Torr, and A. Zisserman, "Objcut: Efficient segmentation using top-down and bottom-up cues," *PAMI*, vol. 32, no. 3, pp. 530–545, 2010.

[4] C.-e. Guo, S.-C. Zhu, and Y. N. Wu, "Towards a mathematical theory of primal sketch and sketchability," in *ICCV*, 2003.

[5] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *PAMI*, vol. 23, no. 11, pp. 1222–1239, 2001.

[6] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *ECCV*, 2006.

[7] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "iCoseg: Interactive co-segmentation with intelligent scribble guidance," in *CVPR*, 2010.

[8] D. Kuettel, M. Guillaumin, and V. Ferrari, "Segmentation propagation in imagenet," in *ECCV*, 2012.

[9] D. Cremers, T. Kohlberger, and C. Schnörr, "Nonlinear shape statistics in Mumford-Shah based segmentation," in *ECCV*, 2002.

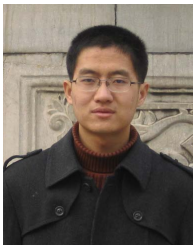
[10] M. Rousson and D. Cremers, "Efficient kernel density estimation of shape and intensity priors for level set segmentation," in *MICCAI*, 2005.

[11] B. Packer, S. Gould, and D. Koller, "A unified contour-pixel model for figure-ground segmentation," in *ECCV*, 2010.

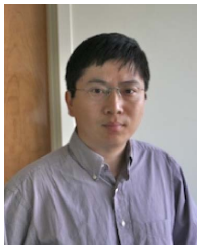
[12] C. Rother, T. Minka, A. Blake, and V. Kolmogorov, "Cosegmentation of image pairs by histogram matching-incorporating a global constraint into MRFs," in *CVPR*, 2006.

[13] D. Batra, D. Parikh, A. Kowdle, T. Chen, and J. Luo, "Seed image selection in interactive cosegmentation," in *ICIP*, 2009.

[14] G. Liu, Z. Lin, Y. Yu, and X. Tang, "Unsupervised object segmentation with a hybrid graph model (hgm)," *PAMI*, vol. 32, no. 5, pp. 910–924, 2010.
[15] A. Joulin, F. Bach, and J. Ponce, "Discriminative clustering for image co-segmentation," in *CVPR*, 2010.
[16] S. Vicente, C. Rother, and V. Kolmogorov, "Object cosegmentation," in *CVPR*, 2011.
[17] G. Kim, E. P. Xing, L. Fei-Fei, and T. Kanade, "Distributed cosegmentation via submodular optimization on anisotropic diffusion," in *ICCV*, 2011.
[18] L. Mukherjee, V. Singh, and J. Peng, "Scale invariant cosegmentation for image groups," in *CVPR*, 2011.
[19] J. C. Rubio, J. Serrat, A. López, and N. Paragios, "Unsupervised co-segmentation through region matching," in *CVPR*, 2012.
[20] A. Joulin, F. Bach, and J. Ponce, "Multi-class cosegmentation," in *CVPR*, 2012.
[21] J. Winn and N. Jojic, "Locus: Learning object classes with unsupervised segmentation," in *ICCV*, 2005.
[22] B. Alexe, T. Deselaers, and V. Ferrari, "Classcut for unsupervised class segmentation," in *ECCV*, 2010.
[23] M. Rubinstein, A. Joulin, J. Kopf, and C. Liu, "Unsupervised joint object discovery and segmentation in internet images," in *CVPR*, 2013.
[24] A. Faktor and M. Irani, "Cosegmentation by composition," in *ICCV*, 2013.
[25] Y. N. Wu, Z. Si, H. Gong, and S.-C. Zhu, "Learning active basis model for object detection and recognition," *IJCV*, vol. 90, no. 2, pp. 198–235, 2010.
[26] Z. Si and S.-C. Zhu, "Learning hybrid image templates (hit) by information projection," *PAMI*, vol. 34, no. 7, pp. 1354–1367, 2012.
[27] N. Ahuja and S. Todorovic, "Extracting texels in 2.1D natural textures," in *ICCV*, 2007.
[28] L. Lin, K. Zeng, X. Liu, and S.-C. Zhu, "Layered graph matching by composite cluster sampling with collaborative and competitive interactions," in *CVPR*, 2009.
[29] S. Lee and Y. Liu, "Skewed rotation symmetry group detection," *PAMI*, vol. 32, no. 9, pp. 1659–1672, 2010.
[30] Y. Hong, Z. Si, W. Hu, S.-C. Zhu, and Y. N. Wu, "Unsupervised learning of compositional sparse code for natural image representation," *Q. Appl. Math.*, in press.
[31] S. Singh, A. Gupta, and A. A. Efros, "Unsupervised discovery of mid-level discriminative patches," in *ECCV*, 2012.
[32] Q. Li, J. Wu, and Z. Tu, "Harvesting mid-level visual concepts from large-scale internet images," in *CVPR*, 2013.
[33] J. Liu and Y. Liu, "Grasp recurring patterns from a single view," in *CVPR*, 2013.
[34] S. Fidler and A. Leonardis, "Towards scalable representations of object categories: Learning a hierarchy of parts," in *CVPR*, 2007.
[35] J. Schlecht, K. Barnard, E. Spriggs, and B. Pryor, "Inferring grammar-based structure models from 3d microscopy data," in *CVPR*, 2007.
[36] L. Zhu, C. Lin, H. Huang, Y. Chen, and A. Yuille, "Unsupervised structure learning: Hierarchical recursive composition, suspicious coincidence and competitive exclusion," in *ECCV*, 2008.
[37] S. Todorovic and N. Ahuja, "Unsupervised category modeling, recognition, and segmentation in images," *PAMI*, vol. 30, no. 12, pp. 2158–2174, 2008.
[38] S. Geman, D. F. Potter, and Z. Chi, "Composition systems," *Q. Appl. Math.*, vol. 60, no. 4, pp. 707–736, 2002.
[39] L. Zhu, Y. Chen, A. Yuille, and W. Freeman, "Latent hierarchical structural learning for object detection," in *CVPR*, 2010.
[40] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *PAMI*, vol. 32, no. 9, pp. 1627–1645, 2010.
[41] J. Dai, Y. N. Wu, J. Zhou, and S.-C. Zhu, "Cosegmentation and cosketch by unsupervised learning," in *ICCV*, 2013.
[42] N. Kumar, P. N. Belhumeur, A. Biswas, D. W. Jacobs, W. J. Kress, I. C. Lopez, and J. V. Soares, "Leafsnap: A computer vision system for automatic plant species identification," in *ECCV*, 2012.
[43] M.-M. Cheng, F.-L. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Repfinder: finding approximately repeated scene elements for image editing," *SIGGRAPH*, vol. 29, no. 4, p. 83, 2010.
[44] J. Feng, J. Zhou, and A. K. Jain, "Orientation field estimation for latent fingerprint enhancement," *PAMI*, vol. 35, no. 4, pp. 925–940, 2013.

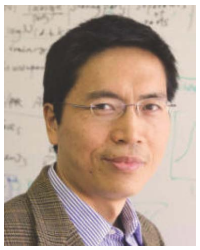


Jifeng Dai received a B.S. degree from Tsinghua University, Beijing, in 2009. He was a visiting student in the Department of Statistics at University of California, Los Angeles, from 2012 to 2013. From 2009 to now, he is working toward a Ph.D. degree in the Department of Automation, Tsinghua University, Beijing. His research interests are in computer vision and pattern recognition. He received Microsoft Research Asia Fellowship Nomination Award, and Scholarship Award for Excellent Doctoral Student granted by Ministry of Education, China.



Ying Nian Wu received his Ph.D. degree in statistics from Harvard University in 1996. He is a professor of statistics at University of California, Los Angeles. His research interest is in statistical modeling and computing, in particular, generative models in computer vision. He received the Marr Prize honorary nominations in 1999 for texture modeling and 2007 for object modeling, respectively.

Jie Zhou (M'01-SM'04) received a Ph.D. degree from Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology (HUST), Wuhan, China. He is a full professor in the Department of Automation, Tsinghua University. His research area includes computer vision, pattern recognition and image processing. In recent years, he has authored more than 100 papers in peer-reviewed journals and conferences. Among them, more than 30 papers have been published in top journals and conferences such as PAMI, T-IP and CVPR. He is an associate editor for International Journal of Robotics and Automation, Acta Automatica and two other journals. Dr. Zhou is a senior member of IEEE and a recipient of the National Outstanding Youth Foundation of China.



Song-Chun Zhu received a Ph.D. degree from Harvard University. He is a professor of Statistics and Computer Science at University of California, Los Angeles. He has published over 160 papers in computer vision, statistical modeling and learning, cognition, and visual arts. He received a number of honors, including the Aggarwal prize from the Intl Association of Pattern Recognition in 2008 for contributions to a unified foundation to computer vision, the Marr Prize in 2003 with Z. Tu et al. for image parsing, twice Marr Prize honorary nominations with Y. N. Wu et al. in 1999 for texture modeling and 2007 for object modeling, a Sloan Fellowship in 2001, a US NSF Career Award in 2001, an US ONR Young Investigator Award in 2001, and the Helmholtz Test-of-Time Award in ICCV 2013 for work on image segmentation. He is a Fellow of IEEE, and served as a general co-chair for CVPR 2012.