J. Vis. Commun. Image R. 34 (2016) 12–27

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation ☆

Hongyuan Zhu [a], Fanman Meng [b], Jianfei Cai [c,*], Shijian Lu [a]

[a] Institute for Infocomm Research, A*STAR, Singapore
[b] School of Electronic Engineering, Univ. of Electronic Science and Technology of China, China
[c] School of Computer Engineering, Nanyang Technological University, Singapore

ARTICLE INFO

Article history:
Received 25 January 2015
Accepted 18 October 2015
Available online 24 October 2015

Keywords:
Image segmentation
Superpixel
Interactive image segmentation
Object proposal
Semantic image parsing
Image cosegmentation
Unsupervised image segmentation
Weakly-supervised image segmentation

ABSTRACT

Image segmentation refers to the process to divide an image into meaningful non-overlapping regions according to human perception, which has become a classic topic since the early ages of computer vision. A lot of research has been conducted and has resulted in many applications. While many segmentation algorithms exist, there are only a few sparse and outdated summarizations available. Thus, in this paper, we aim to provide a comprehensive review of the recent progress in the field. Covering 190 publications, we give an overview of broad segmentation topics including not only the classic unsupervised methods, but also the recent weakly-/semi-supervised methods and the fully-supervised methods. In addition, we review the existing influential datasets and evaluation metrics. We also suggest some design choices and research directions for future research in image segmentation.

1. Introduction

Human can quickly localize many patterns and automatically group them into meaningful parts. Perceptual grouping refers to human's visual ability to abstract high-level semantic information (e.g. object classes, shape, scene geometry, physical support, etc.) from low-level image cues (e.g. intensity, color and texture, etc.) without any specific knowledge of image content. Discovering the working mechanisms under this ability has long been studied by the cognitive scientists since 1920s [1]. With the development of modern computer, computer scientists ambitiously want to equip the computer with the perceptual grouping ability given many promising applications, which lays down the foundation for image segmentation, and has been a classical topic since early years of computer vision.

Image segmentation, as a basic operation in computer vision, refers to the process to divide a natural image into some non-overlapped meaningful regions (e.g., objects or parts). However, what makes an "object" or a part "meaningful" can be ambiguous. An "object" can be referred to a "thing" (e.g. a cup, a cow), a kind of texture (e.g. wood, rock) or even a "stuff" (e.g. a building or a forest). Sometimes, an "object" can also be part of other "objects". Lacking a clear definition of the "object" makes image segmentation a challenging and ill-posed problem. Fig. 1 illustrates such an example, where different human subjects have different ways in interpreting the objects. In this sense, what makes a 'good' segmentation needs to be properly defined.

Another challenge which makes "object" segmentation difficult is how to effectively represent the "object". When human perceives an image, elements in the brain will be perceived as a whole, but most images in computers are currently represented based on low-level features such as color, texture, curvature, and convexity. Such low-level features reflect objects' local properties, which are difficult to capture global object information (e.g. shape).

Early research in human perception has provided some useful guidelines for developing segmentation algorithms. For example, cognition study [3] shows that human vision system views part boundaries at those with negative minima of curvature, and the part salience depends on three factors: the relative size, the boundary strength and the degree of protrusion. Gestalt theory and other psychological studies have also developed various principles reflecting human perception, e.g. (1) human tends to group elements which have similarities in color, shape or other properties; (2) human favors linking contours whenever the elements of the pattern establish an implied direction, etc.

A lot of research has been conducted on image segmentation. Existing methods can be classified into three major categories:

☆ This paper has been recommended for acceptance by M.T. Sun.

* Corresponding author.
E-mail addresses: zhuh@i2r.a-star.edu.sg (H. Zhu), fmmeng@uestc.edu.cn (F. Meng), asjfcai@ntu.edu.sg (J. Cai), slu@i2r.a-star.edu.sg (S. Lu).

http://dx.doi.org/10.1016/j.jvcir.2015.10.012
1047-3203/

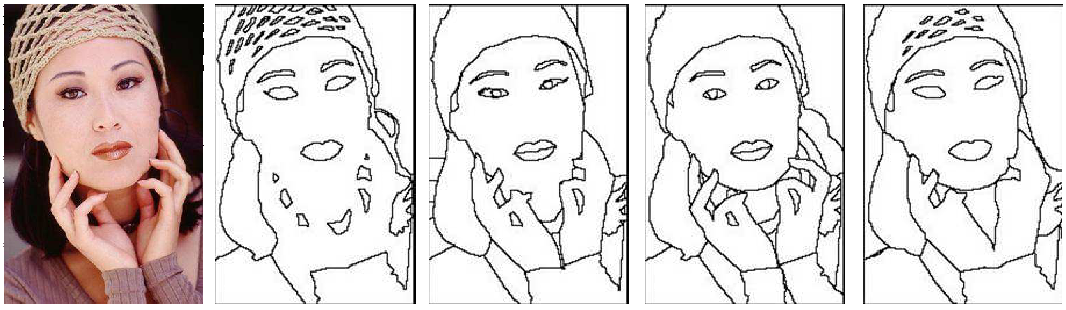

Fig. 1. An image from Berkeley segmentation dataset [2] with hand labels from different human subjects, which demonstrates the variety of human perception.

unsupervised methods, weakly-/semi-supervised methods and fully-supervised methods, with an increasing level of supervision. Their relationship is illustrated in Fig. 2. These methods are inherently correlated with each other, and there are no clear boundaries between them. Unsupervised methods segment an image into homogeneous meaningful regions with no human intervention. Early techniques focus on local region merging and splitting and borrowed some methods from clustering community [9,10]. Recent techniques seek to optimize some global criteria [11–17]. Weakly-/semi-supervised methods make use of coarse or partial annotations to segment the un-annotated image data. Interactive segmentation methods [18–20] as one sub-category of the weakly-supervised methods which utilize user input, have been applied in some commercial products such as Microsoft Office and Adobe Photoshop. With the emergence of large-scale image databases, such as the ImageNet [21] and personal photo streams on Flickr, the cosegmentation methods [22–29] which can extract recurring objects from a set of images, have attracted increasing attentions in these years. Supervised methods build up the model of object-of-interest for specific application with annotated data in fine-grained form. Hence, supervised methods can achieve supreme performance for certain object classes and tasks. Recent substantial development of image classification [30], object detection [31], superpixel segmentation [32] and 3D scene recovery [33] have boosted the research in supervised methods, especially on scene parsing [34–39].

Image segmentation has proved to be useful in many image processing, computer vision and multimedia tasks, where some sample applications are illustrated in Fig. 3. For example, unsupervised segmentation has been applied in image annotation [40] by decomposing an image into several blobs corresponding to objects. Unsupervised methods have also been widely used in hyperspectral/multi-spectral satellite images for object detection [41]. Superpixel segmentation, which transforms millions of pixels into hundreds or thousands of homogeneous regions [32,11], has been applied to reduce some complex vision tasks' complexity and improve their speed and accuracy, such as estimating dense correspondence field [42], scene parsing [43] and body model estimation [44]. Interactive methods have been used in image montage [4], colorization [5] and other multimedia manipulation tasks. They also play important roles in medical applications such as organ reconstruction [7]. Foreground region pools, or so called region proposals [45–47], have been applied to facilitate object detection, which provides better localization than sliding windows. Supervised scene understanding has been used in auto-vehicles [6] and object pop-up [8]. The segmentation techniques developed for images, such as Mean Shift [11] and Normalized Cut [13] have also been applied in other areas such as data clustering and density estimation.

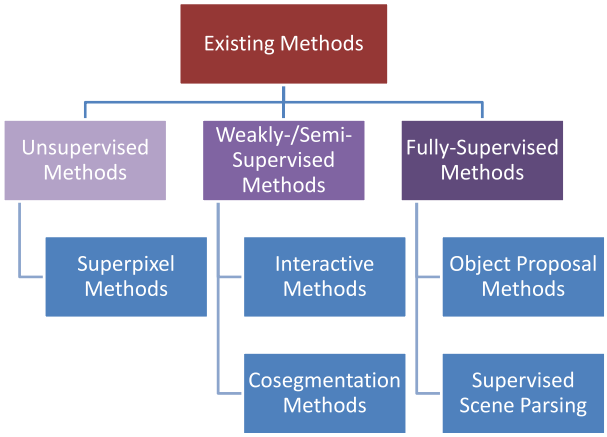

Fig. 2. Existing methods can be classified into three main categories in purple colors. An increasing intensity of the purple color indicates the increasing level of supervision. The rectangles in blue color are the sub-categories of the main categories. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Although image segmentation research has been evolving for a long time, the challenges ranging from feature representation to model design and optimization still hinder the further performance improvement toward human perception. Hence, it is necessary to periodically give a thorough and systematic review of existing methods, especially the recent ones, to summarize what have been achieved, where we are now, what knowledge and lessons can be shared and transferred between different communities, and what are the directions and opportunities for future research. To our surprise, there are only a few sparse and outdated reviews on segmentation literature. There is no comprehensive review to cover broad areas of segmentation topics including not only the classic unsupervised methods, but also the recent development in weakly-/semi-supervised methods and fully-supervised methods, which are critically and exhaustively reviewed in this paper in Sections 2–4. In addition, we also review the existing influential datasets and evaluation metrics in Section 5. Finally, we discuss some popular design choices and some potential future directions in Section 6, and conclude the paper in Section 7.

2. Unsupervised methods

Unsupervised methods group local pixels which are homogeneous in low-level features (e.g. color, texture or curvature) into non-overlapped larger regions that may potentially correspond to objects or their parts without any training examples. Hence, unsupervised methods do not have any explicit object models. They achieve grouping by clustering those features based on fitting mixture models, mode shifting [11] or graph partitioning [12–15]. In addition, the continuous or variational techniques [16,17] have also been used to segment images into regions. The regions

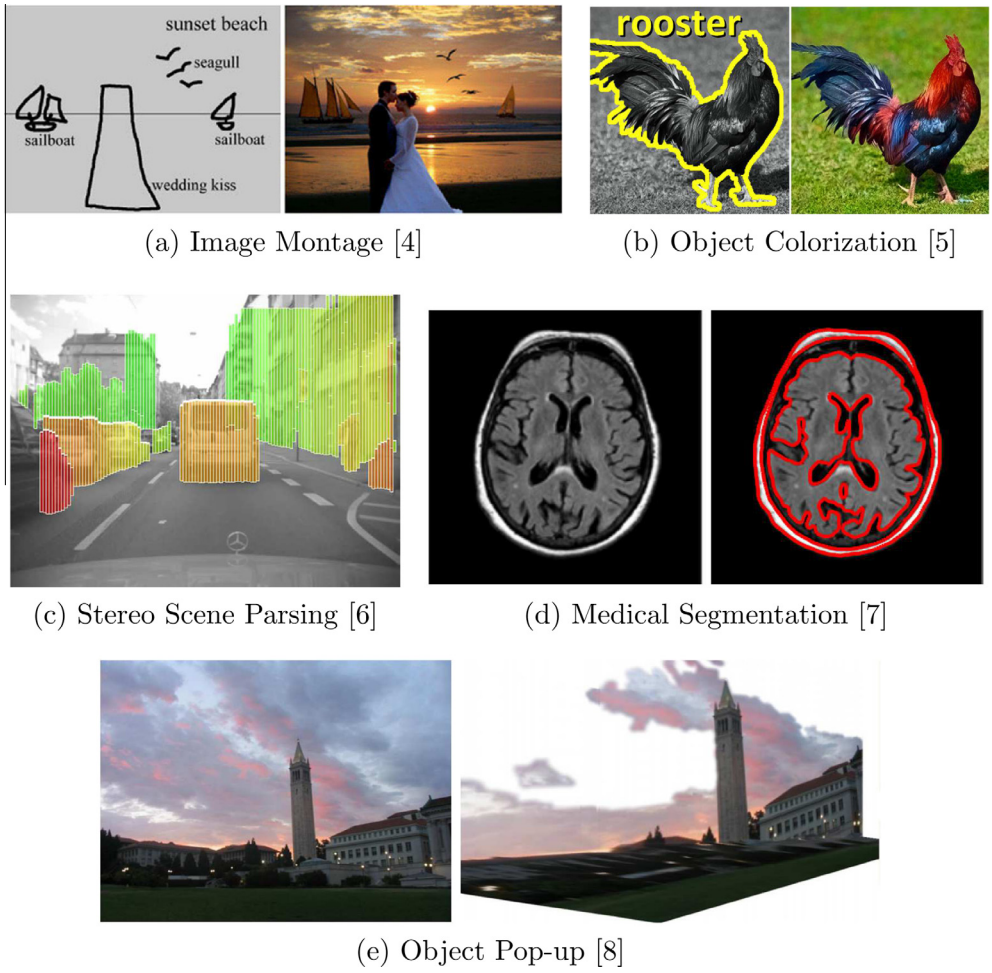

Fig. 3. Sample applications of image segmentation. Images are reproduced from corresponding references.

produced by the unsupervised methods help transform millions of image pixels into a much smaller number of mid-level representations, and facilitate higher-level analysis and understanding tasks.

Below we give a brief summary of some popular unsupervised methods due to their reasonable performance and publicly available implementation. We divide these methods into two major categories: *discrete methods* and *continuous methods*, where the former category considers an image as a fixed discrete grid while the latter one treats an image as a continuous surface.

2.1. Discrete methods

The discrete methods can be further divided into two subcategories: clustering based methods, and graph based methods.

(a) Clustering based methods: Early unsupervised methods are borrowed from the feature space analysis methods in clustering community, which maps an image's pixels to the feature vector space, then apply either *parametric* or *non-parametric* approaches to fit the clusters in this space. Most clustering methods operate on a feature space which consists of local features, hence they are sensitive to the changes in local statistics, which tend to segment an image into a large number of small regions.

Parametric approaches require a prior knowledge of the region/cluster number (including those which use optimization to determine the region number) and the cluster shape (sphere or elliptical), which models the images using a few fixed parameters. Given that cluster number and distributions of the clusters are known, parametric methods works effectively. However, these assumptions are not generally satisfied for high dimensional pixel features, hence parametric methods' performance is unsatisfactory for objects with complex feature space shape.

K-means is among the simplest parametric method. Given k initial centers which can be randomly selected, K-means first assigns each sample to one of the centers based on their feature space distance and then the centers are updated. These two steps iterate until the termination conditions are met. *Mixtures of Gaussian* is similar to K-means. Within Mixtures of Gaussian, each cluster center is now replaced by a covariance matrix, therefore can better handle elliptical clusters. Recently, Rao et al. [48] extended the Gaussian mixture models by encoding the texture and boundary using minimum description length theory and achieved better performance.

Non-parametric approaches are different from the parametric methods which have assumptions over the cluster number and feature distributions. The non-parametric methods can automatically determine the cluster number and the modes in the feature space, hence can better fit the clusters in the arbitrarily shaped feature space.

Region merging and splitting methods merge or split pixels/image regions according to some proximity measures. Watershed [14] segments an image into the catchment and basin by flooding the morphological surface at the local minimum and then constructing ridges at the places where different components meet. Region splitting [9] models the image as a histogram, and then a

threshold to split the histogram peaks are decided to recursively split the regions into sub-regions. Region merging starts from blocks or regions, then adjacent regions are recursively merged according to the edge strengths between them [10,49,50]. The region merging (splitting) processes tend to be computationally expensive and the definition of stopping criteria for merging (splitting) is not straightforward.

Mean-shift [11] is a classic non-parametric method based on density estimation which models the features space as a probability density functions (p.d.f). The clusters of the feature space correspond to the modes of the p.d.f. Finding modes of the p.d.f automatically determine the clusters without explicitly determining the threshold. Assume a data point x are drawn from some probability function, whose density can be estimated by convolving the data with a fixed kernel of width h:

$$f(x) = \sum_i K(x - x_i) = \sum_i k\left(\frac{||x - x_i||^2}{h^2}\right) \tag{1}$$

where x_i is a near-by sample and $k(\cdot)$ is the kernel function [51]. After the density function is estimated, mean shift uses a multiple restart gradient descent method which starts at some initial guess y_k, then the gradient direction of $f(x)$ is estimated at y_k and uphill step is taken in the direction [51]. During the mean-shift procedure, the current mode y_k is replaced by its locally weighted mean:

$$y_{k+1} = y_k + m(y_k) \tag{2}$$

Final segmentation is formed by grouping pixels whose converge points are closer than h_s in the spatial domain and h_r in the range domain, and these two parameters are tuned according to the requirement of different applications.

(b) Graph based methods model an image as a graph $G = \{V, E\}$, where V is the set of pixels/regions, E is the set of the edges connecting the pixels/regions, which reflect the similarities between the pixels/regions. Then the methods segment the image according some optimization function defined on the graph. As the graph conveys global image information, graph based methods which optimize global functions are less sensitive to the changes in local features.

Graph based region merging [12] advocated a method which uses a *relative* dissimilar measure to produce segmentation which optimizes a global grouping metric, which is different from the region merging methods which use *fixed* merging rules as introduced in clustering methods. The method maps an image to a graph with a 4-neighbor or 8-neighbor structure. The pixels denote the nodes, while the edge weights reflect the color dissimilarity between the nodes. Initially each node forms their own component. The internal difference $Int(C)$ is defined as the largest weight in the minimum spanning tree of a component C. Then the weights are sorted in an ascending order. Two regions C_1 and C_2 are merged if the in-between edge weight is less than $min(Int(C_1) + \tau(C_1), Int(C_2) + \tau(C_2))$, where $\tau(C) = k/|C|$ and k is a coefficient that is used to control the component size. Merging stops when the difference between components exceeds the internal difference.

Normalized Cut: Many methods generate the segmentation based on local image cues, and hence they could produce trivial regions because the low-level features are sensitive to the lighting and perspective changes. In contrast, Normalized Cut [52] finds a segmentation via splitting the affinity graph which encodes the global image information, i.e. minimizing the *Ncut* value between different clusters:

$$Ncut(S_1, S_2, \ldots, S_k) := \frac{1}{2}\sum_{i=1}^{k}\frac{W(S_i, \bar{S}_i)}{vol(S_i)} \tag{3}$$

where $S_1, S_2, \ldots, S_k$ form a k-partition of a graph, $\bar{S}_i$ is the complement of S_i, $W(S_i, \bar{S}_i)$ is the sum of the boundary edge weights of S_i, and $vol(S_i)$ is the sum of the weights of all edges attached to vertices in S_i. The basic idea here is that big clusters have large $vol(S_i)$ and minimizing *Ncut* encourages all $vol(S_i)$ to be about the same, thus achieving a "balanced" clustering.

Finding the normalized cut is an NP-hard problem. Usually, an approximate solution is sought by computing the eigenvectors v of the generalized eigenvalue system $(D - W)v = \lambda Dv$, where $W = [w_{ij}]$ is the affinity matrix of an image graph with w_{ij} describing the pairwise affinity of two pixels and $D = [d_{ij}]$ is the diagonal matrix with $d_{ii} = \sum_j w_{ij}$.

The segmentation is achieved by recursively bi-partitioning the graph using the non-zero eigenvalue's eigenvectors [13] or clustering of a set of eigenvectors [53]. For the computational efficiency purpose, spectral clustering requires the affinity matrix to be sparse which limits its applications. Recent work of Cour et al. [54] solved this limitation by defining the affinity matrix at multiple scales and then setting up cross-scale constraints which achieved better result. In addition, Arbelaez et al. [55] combined the contour information from the eigenvectors and the oriented contour response at local pixel to form the state-of-the-art contour detectors, the watershed is applied to the contour map to oversegment the image into small regions, and then these regions are merged according to the pair-wise contour strength, which is called the gPb-OWT-UCM methods.

2.2. Continuous methods

Continuous methods [17,16,56–58] have also been applied for segmenting an image into regions. These methods treat an image as a continuous surface instead of a fixed discrete grid, hence can avoid the grid bias artefacts of discrete methods and produce visually more pleasing results.

The Mumford–Shah (MS) model partitions an image by minimizing the functional which encourages homogeneity within each region as well as sharp piecewise regular boundaries. The MS functional is defined as

$$F_{MS}(I, C) = \int_\omega |I - I_0|^2 dx + \mu \int_{\omega\setminus C} |\nabla I|^2 dx + \nu H^{N-1}(C) \tag{4}$$

for any observed image I_0 and any positive parameters $\mu,\ \nu$, where I corresponds to a piecewise smooth approximation of I_0, C represents the boundary contours of I and its length is given by Hausdorff measure $H^{N-1}(C)$. The first term of (4) measures the fidelity of I with respect to the given data I_0, the second term regularizes the function I to be smooth inside the region $\omega \setminus C$ and the last term regularizes the discontinuity set C to be smooth.

Minimizing the Mumford–Shah model is not easy, many variants have been proposed to approximate the functional [59–62]. Among them, Vese–Chan [63] proposed to approximate the term $H^{N-1}(C)$ by the lengths of region contours, which provides the model of *active contour without edges*. By assuming the region is piecewise constant, the model is further simplified to the continuous Potts model, which has convexified solvers [57,58].

Active contour/snake model [64] detects objects by deforming a contour curve C towards the sharp image edges. The evolution of parametric curve $C(p) = (x(p), y(p)), p \in \{0, 1\}$ is driven by minimizing the functional:

$$F(C) = \alpha \int_0^1 \left|\frac{\partial C(p)}{\partial p}\right|^2 dp + \beta \int_0^1 \left|\frac{\partial^2 C}{\partial p^2}\right|^2 dp + \lambda \int_0^1 f^2(I_0(C))dp \tag{5}$$

where the first two terms enforce smoothness constraints by making the snake act as a membrane and a thin plate correspondingly, and the sum of the first two terms makes the *internal energy*. The

third term, called *external energy*, attracts the curve toward the object boundaries by using the edge detecting function

$$f(I_0) = \frac{1}{1+\gamma|\nabla(I_0 * G_\sigma)|^2} \quad (6)$$

where γ is an arbitrary positive constant and $I_0 * G_\sigma$ is the Gaussian smoothed version of I_0. The energy function is non-convex and sensitive to initialization. To overcome the limitations, Osher and Sethian [17] proposed the *level set method*, which implicitly represents curve C by a higher dimension ψ, called the level set function. Moreover, Bresson et al. [56] proposed the convex relaxed active contour model which can achieve desirable global optimal.

2.3. Superpixel

Superpixels are homogeneous regions which are smaller than objects or their parts. In the seminal work of Ren and Malik [65], they argued and justified that superpixels are more natural and efficient representations than the pixels. The benefit of using superpixels are due to the small region number and their relatively large spatial support, which greatly reduce the model complexity and increase the robustness of subsequent applications. There are different paradigms to produce superpixels.

- Existing unsupervised methods can be directly adapted to produce superpixels by tuning the parameters, e.g. Mean-Shift, Normalized Cut (by increasing cluster number), Graph Based Merging (by controlling the regions size) and Mean-Shift/Quick-Shift (by tuning the kernel size or changing mode drifting style). However, they generate superpixel at slow speed due to their slow optimization pipelines (e.g. Normalized Cut). They could also produce superpixels of irregular shape due to lacking of shape control (e.g. Graph Based Merging, Mean-shift), which can reduce the efficiency of feature extraction.
- Some recent methods produce much faster and better superpixel segmentation by shifting optimization scope from the whole image to regularly placed seed regions, and then adjusting these regions' boundaries to snap to the salient object contours with shape control. TurboPixel [66] deforms the initial spatial grid to compact and regular regions by using geometric flow (or localized level set) which is directed by local gradients. Wang et al. [67] also adapted geodesic flows by computing geodesic distance among pixels to produce adaptive superpixels, which have higher density in high intensity or color variation regions while having larger superpixels at structure-less regions. Veksler et al. [68] proposed to place overlapping patches at the image, and then assigned each pixel to a local region by using graph-cut, which is to be introduced in Section 3. Zhang et al. [69] further studied in this direction by using a pseudo-boolean optimization which achieves faster speed. Achanta et al. [32] introduced the SLIC algorithm which greatly improves the superpixel efficiency. SLIC starts from the initial regular grid of superpixels, grows superpixels by estimating each pixel's distance to its cluster center localized nearby, and then updates the cluster centers, which is essentially a localized K-means. SLIC can produce superpixels at 5 Hz.
- There are also some new formulations for over-segmentation. Liu et al. [70] proposed a new graph based method, which can maximize the entropy rate of the cuts in the graph with a balance term for compact representation. Although it outperforms many methods in terms of boundary recall measure, it takes about 2.5 s to segment an image of size 480×320. Van den Berge et al. [71] proposed the fastest superpixel method-SEED. SEED uses multi-resolution image grids as initial regions. For each image grid, they define a color histogram based entropy term and an optional boundary term. Instead of using iterated refinement as SLIC, which needs to repeatedly compute distances, SEEDs uses Hill-Climbing to move coarser-resolution grids, and then refines region boundary using finer-resolution grids. In this way, SEEDs can achieve real time superpixel segmentation at 30 Hz.

3. Weakly-/semi-supervised methods

Image segmentation is expected to produce regions which match human perception. It is difficult for unsupervised methods to produce object regions using low-level cues without any high-level information. Incorporating high-level information makes the ill-posed segmentation problem better defined. However, such information often requires expensive manual or time cost. Weakly-/semi-supervised methods allow incorporating small amount of prior knowledge by using labeled data in a weak or coarse form, e.g. by partially providing labels for a few pixels (e.g. interactive methods) or by manually picking out images containing common objects (e.g. cosegmentation methods). Weakly supervised methods involve a self-training procedure by using a few coarsely labeled training examples to train a model to segment an image/a set of images, and then use the thought-to-be positive examples yielded to re-train the model. Semi-supervised methods not only make use of small amount of labeled data but also additionally incorporate the unlabeled data to train the model, which can better model the inherent structure of the data.

3.1. Interactive methods

In general, an interactive method has the following pipeline: (1) the user labels a few pixels using the provided interface as initial constraints; (2) then the method produces the best result under current constraints; (3) based on the result, user provides further constraints, and then go back to step 2 to refine the result. The processes repeat until the result is satisfactory. The interface of interactive methods provides a bridge between the user and the algorithm to convey his prior knowledge (see Fig. 4 for examples). According to the methodology, existing interactive methods can be classified into three groups: (i) *contour tracking approaches* [17,72,73,20,74], (ii) *label propagation approaches* [75–77] and (iii) *local optimization approaches* [78,79]. There are already some surveys on interactive segmentation methods [80,81,75], and thus this paper will act as a supplementary to them, where we discuss recent influential literature not covered by the previous surveys. In addition, to make the manuscript self-contained, we will also give a brief review of some classical techniques.

(1) Contour tracking approaches are one type of the earliest interactive segmentation methods. In a typical contour tracking method, the user first places a contour close to the object boundary, and then the contour will evolve to snap to the nearby salient object boundary. Contour tracking methods require a large number of user input when handling highly textured or jaggy objects, and they are also sensitive to the initialization, which are easy to get stuck in bad local optimum.

The *Live-wire/intelligent scissors* method [72,73] starts contour evolving by building a weighted graph on the image, where each node in the graph corresponds to a pixel, and directed edges are formed around pixels with their closest four neighbors or eight neighbors. The local cost of each direct edge is the weighted sum of Laplacian zero-cross, gradient magnitude and gradient direction. Then given the seed locations, the shortest path from a seed point p to a certain seed point s is found by using Dijkstra method. Essentially, Live-Wire minimizes a local energy function, which is easy to get stuck in bad local minimum. The active contour method introduced in Section 2.2 has also been applied in interactive

Fig. 4. Examples of interactive image segmentation using bounding box (a & b) [18] or scribbles (c & d) [19].

segmentation. Nguyen et al. [20] recently employed the convex active contour model to refine the object boundary produced by other interactive segmentation methods and achieved satisfactory segmentation results. The shape prior has also been incorporated in the active contour to allow the model to better snap to desirable structure [82]. Liu and Yu [74] proposed to use the level set function [17] to track the zero level set of the posterior probabilistic mask learned from the user provided labels.

(2) Label propagation approaches are more popular in recent literature due to their reliance on global optimization methods or local optimization methods with a large spatial support, hence are less sensitive to local optimum. The basic idea of these approaches are to propagate the user-provided initial labels to unlabeled pixels using either global optimization (such as GraphCut [76] or RandomWalk [77]) or local optimization (such as bilateral filtering with a large kernel).

GraphCut based methods [76,18,83] model the pixel labeling problem in Markov Random Field. Their energy functions can be generally expressed as

$$E(X) = \sum_{x_i \in X} E_u(x_i) + \sum_{x_i, x_j \in \mathcal{N}} E_p(x_i, x_j) \tag{7}$$

where $X = \{x_1, x_2, \ldots, x_n\}$ is the set of random variables defined at each pixel, which can take either foreground label (1) or background label (0); $\mathcal{N}$ defines a neighborhood system, which is typically 4-neighbor or 8-neighbor. The first term $\sum_{x_i \in X} E_u(x_i)$ in (7) is the unary potential, and the second term $\sum_{x_i, x_j \in \mathcal{N}} E_p(x_i, x_j)$ is the pairwise potential. In [76], the unary potential is evaluated by using an intensity histogram. Later in GrabCut [18], the unary potential is derived from the two Gaussian Mixture Models (GMMs) to model the color background and foreground regions respectively. Li et al. [83] further proposed the LazySnap method, which extends GrabCut by using superpixels and including an interface to allow user to adjust the result at the low-contrast and weak object boundaries. One limitation of GrabCut is that it favors short boundary due to the energy function design. Recently Kohli et al. [84] proposed to use a conditional random field with multiple-layered hidden units to encode boundary preserving higher order potential, which has efficient solvers and therefore can capture thin details that are often neglected by classical MRF model.

MRF/CRF model provides a unified framework to combine multiple information, hence various prior knowledge has been incorporated. One typical assumption on objects is that they are compactly clustered in spatial space, instead of distributing around. Such spatial constraint can be captured by measuring the geodesic distance from the pixels to foreground (background) seeds. Then the geodesic distance can be designed as a kind of data term in the energy function as in [85]. Later, Bai and Sapiro [86] further extended the solution to soft matting problem. Zhu et al. [87] also incorporated the Bai's geodesic distance as one type of unary object potential to produce unsupervised object segmentation, which achieves improved results.

Another common assumption is that most objects' contour form a convex shape. For example, a star shape is defined with respect to a center point c. An object has a star shape if for any point p inside the object, all points on the straight line between the center c and p also lie inside the object. Veksler [88] formulate such constraint by penalizing different labeling on the same line, which such formulation can only work with single convex center. Gulshan et al. [89] proposed to use geodesic distance transform [79] to compute the geodesic convexity from each pixel to the star center, which works on objects with multiple start centers. Other similar connectivity constraint has also been studied in [90].

The *RandomWalk* model [77] provides another pixel labeling framework, which has been applied to many computer vision problems, such as segmentation, denoising and image matching. With notations similar to those for GraphCut in (7), RandomWalk starts from building an undirected graph $G = (V, E)$, where V is the set of vertices defined at each pixel and E is the set of edges which incorporate the pairwise weight W_{ij} to reflect the probability of a random walker to jump between two nodes i and j. The *degree* of a vertex is defined as $d_i = \sum_j W_{ij}$.

Given the weighted graph, a set of user scribbled nodes V_m and a set of unmarked nodes V_u, such that $V_m \cup V_u = V$ and $V_m \cap V_u = \Phi$, the RandomWalk approach solves a large scale sparse linear system to assign each node $i \in V_u$ a probability x_i that a random walker starting from that node first reaches a marked node. The final segmentation is formed by thresholding x_i. Yang et al. [19] further proposed a constrained RandomWalk that is able to incorporate different types of user inputs as additional constraints such as hard and soft constraints to provide more flexibility for users.

(3) Local optimization approaches: Many label propagation methods use specific solvers such as graph cut or random walk to get the most likely solution (or segmentation). However, such global solvers do not scale well with image size. Hosni et al. [78] demonstrated that by filtering the unary cost volume by using fast edge preserving filters (such as Joint Bilateral Filter [91], Guided Filter [92]) with a large kernel (e.g. 8×8, Cross-map filter [93], etc.) and then using Winner-Takes-All method to take the most likely labels, they can achieve comparable or better results than global optimized models. More importantly, such cost-volume filtering approaches can achieve real time performance, e.g. 2.85 ms to filter an 1 M-pix image on a Core 2 Quad 2.4 GHz desktop. Recently, Criminisi et al. [79] proposed to use geodesic distance transform to filter the cost volume, which can produce results comparable to the global optimization methods and can better capture the edges at the weak boundaries.

3.2. Image cosegmentation

Cosegmentation aims at extracting common objects from a set of images (see Fig. 5 for examples). It gives very weak prior that the images contain the same objects for automatic object segmentation. Since it does not need any pixel level or image level object information, it is suitable for large scale image dataset

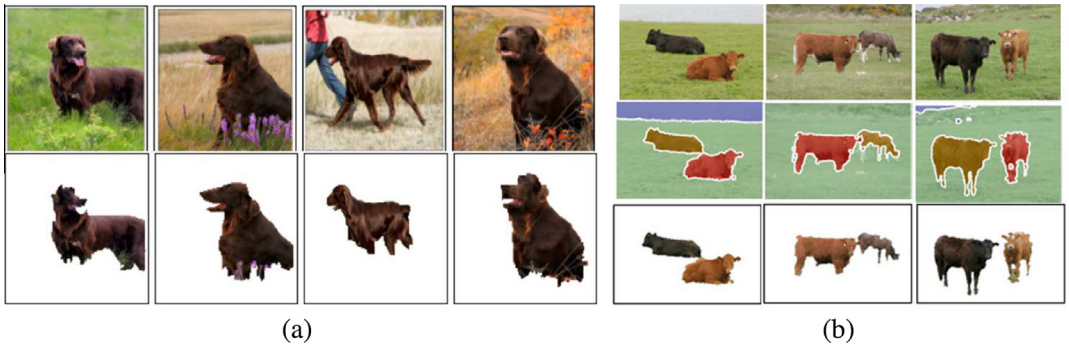

Fig. 5. (a) An example of simultaneously segmenting one common foreground object from a set of image [27]. (b) An example of multi-class cosegmentation [23].

segmentation and has many practical applications, which attracts much attention recently. Cosegmentation was first introduced by Rother et al. [94] in 2006. After that, many cosegmentation methods have been proposed [22–29]. The existing methods can be roughly classified into three categories. The first one is to extend the existing single image based segmentation models to solve the cosegmentation problem, such as MRF methods, RandomWalk methods and active contour based methods, which have been introduced in Section 3.1. The second one is to design new cosegmentation models, such as formulating the cosegmentation as clustering problem, graph theory based proposal selection problem, metric rank based representation. The last one is to solve new emerging cosegmentation needs, such as multiple class/foreground object cosegmentation, large scale image cosegmentation and web image cosegmentation.

(1) Cosegmentation by extending single-image segmentation models: A straight-forward way for cosegmentation is to extend the existing classical single image based segmentation models in Section 3.1 to handle multiple images. In general, the extended models can be represented as

$$E = E_s + E_g \tag{8}$$

where E_s is the single image segmentation term, which guarantees the smoothness and the distinction between foreground and background in each image, and E_g is the cosegmentation term, which focuses on evaluating the consistency between the foregrounds among the images. Only the segmentation of common objects can result in small values of both E_s and E_g. Thus, the cosegmentation is formulated as minimizing the energy in (8).

Many classical segmentation models have been used to form E_s, e.g. using MRF segmentation models [94,29] as

$$E_s^{MRF} = E_u^{MRF} + E_p^{MRF} \tag{9}$$

where E_u^{MRF} and E_p^{MRF} are the conventional unary potential and the pairwise potential as in Eq. (7). The cosegmentation term E_g is used to penalize the multiple foreground color histograms' inconsistency. Various cosegmentation terms and their minimization methods have been carefully designed in MRF based cosegmentation models.

In particular, Rother et al. [94] evaluated the consistency by ℓ_1 norm, Adding the ℓ_1 evaluation makes the minimization quite challenging. An approximation method called submodular–supermodular procedure has been proposed to minimize the model by max-flow algorithm. Mukherjee et al. [29] replaced ℓ_1 with squared ℓ_2 evaluation. The ℓ_2 has several advantages, such as it allows for relaxing the minimization to LP problem and using Pseudo-Boolean optimization method for minimization. But it is still an approximation solution. In order to simplify the model minimization, Hochbaum and Singh [95] used reward strategy rather than the penalty strategy. The energy generated with MRF model is proved to be submodular and can be efficiently solved by GraphCut algorithm. Rubio et al. [25] evaluate the foreground similarities by high order graph matching, which is introduced into MRF model to form the global term. Batra et al. [96] firstly proposed an interactive cosegmentation, where an automatic recommendation system was developed to guide the user to scribble the uncertain regions for cosegmentation refinement. Apart from MRF segmentation model, Collins et al. [97] extended RandomWalk model in Section 3.1 to solve cosegmentation, which results in a convex minimization problem with box constraints and a GPU implementation. In [98], active contour based segmentation is extended for cosegmentation, which consists of the foreground consistencies across images and the background consistencies within each image. Due to the linear similarity measurement, the minimization can be resolved by the level set technique.

(2) New cosegmentation models: The second category tries to solve cosegmentation problem using new strategies rather than extending existing single segmentation models. For example, by treating the common object extraction task as a common region clustering problem, the cosegmentation problem can be solved by clustering strategy. Joulin et al. [26] treat the cosegmentation as a discriminative clustering problem, and trained a supervised classifier based on the foreground and background labels to see if the given labels are able to result in maximal separation of the foreground and background classes.

By representing the region similarity relationships as edge weights of a graph, graph theory has also been used to solve cosegmentation. Kim et al. [99] solved cosegmentation by dividing the images into hierarchical superpixel layers and describing the relationship of the superpixels using graph. Affinity matrix considering intra-image edge affinity and inter-image edge affinity is constructed. The cosegmentation can then be solved by spectral clustering strategy. In [100], by representing each image as a set of object proposals, a random forest regression based model is learned to select the object from the backgrounds. Meng et al. [101] constructed a directed graph to describe the foreground relationship by only considering the neighboring images. The object cosegmentation is then formulated as s shortest path problem, which can be solved by dynamic programming.

Some methods try to learn a model for common objects, which is then used to extract the common objects. Sun and Ponce [102] solved cosegmentation by learning discriminative part detectors of the object. The discriminative parts are learned based on the fact that the part detector of the common objects should more frequently appear in positive samples than negative samples. Dai et al. [103] proposed co-skech model by extending the active basis model to solve cosegmentation problem. A deformable shape template represented by codebook is generated to align and extract the common object. There are also some other strategies. Faktor and

Irani [104] solved cosegmentation based on the similarity of composition, where the likelihood of the co-occurring region is high if it is non-trivial and has good match with other compositions. Mukherjee et al. [28] evaluated the foreground similarity by forcing low entropy of the matrix comprised by the foreground features, which can handle the scale variation very well.

(3) New cosegmentation problems: Many applications require the cosegmentation on a large-scale set of images, which is extremely time consuming. Kim et al. [23] solved the large-scale multi-class cosegmentation by temperature maximization on anisotropic heat diffusion (called CoSand). Wang and Liu [105] proposed a semi-supervised learning based method for large scale image cosegmentation with very limited training data. Zhu et al. [106] proposed the first method which uses search engine to retrieve similar images to the input image to analyse the object-of-interest information, and then used the information to cut-out the object-of-interest. Rubinstein et al. [107] observed that there are always noise images (which do not contain the common objects) from web image dataset, and proposed a cosegmentation model to avoid the noise images. The main idea is to match the foregrounds using SIFT flow to decide which images do not contain the common objects.

Applying cosegmentation to improve image classification is an important application of cosegmentation. Chai et al. [108] proposed a bi-level cosegmentation method, and used it for image classification. It performs cosegmentation by using bottom-level salient foreground obtained by GrabCut algorithm to initialize the holistic foreground model and the propagate the information with a discriminative classification. Later, a TriCos model [109] containing three levels were further proposed, including image-level, dataset-level, and category-level segmentation, which outperforms the bi-level model. Kuttel et al. [110] proposed to propagate segmentation masks in ImageNet by exploiting the hierarchical structures of the database. Recently, Meng et al. [111] proposed to conduct multiple image groups cosegmentation instead of single image group cosegmentation, which is motivated based on the observation that there exists unbalance cosegmentation difficulty for different groups (e.g., it is easy to extract common objects from some image group while others are still difficult), hence can be used to improve the holistic cosegmentation results.

The problem of extracting multiple objects from personal albums is called "*Multiple Foreground Cosegmentation*" (MFC) (see Fig. 6), where the album contains multiple objects-of-interest and each image in the album contains a subset of them. Kim and Xing [22] proposed the first method to handle MFC problem. Their method starts from building appearance models for the object-of-interest. Then they used beam search to find proposal candidates for each foreground. Finally, the candidates are seamed into non-overlap regions by using dynamic programming. Ma and Latecki [112] formulated the multiple foreground cosegmentation as semi-supervised learning (graph transduction learning), and introduced connectivity constraints to enforce the extraction of connected regions. A cutting-plane algorithm was designed to efficiently minimize the model in polynomial time.

Kim and Ma's methods hold an implicit constraint on objects using low-level cues, and therefore their method might assign labels of "stuff" (grass, sky) to "thing" (people or other objects). Zhu et al. [113] proposed a principled CRF framework which explicitly expresses the object constraints from object detectors and solves an even more challenging problem: *multiple foreground recognition and cosegmentation* (MFRC). They proposed an extended multiple color-line based object detector which can be on-line trained by using user-provided bounding boxes to detect objects in unlabeled images. Finally, all the cues from bottom-up pixels, middle-level contours and high-level object detectors are integrated in a robust high-order CRF model, which can enforce the label consistency among pixels, superpixels and object detections simultaneously, produce higher accuracy in object regions and achieve state-of-the-art performance for the MFRC problem. Later, Zhu et al. [114] extended the framework to the challenging multiple human identification and cosegmentation problem, and proposed a novel shape cue which uses geodesic filters [79] and joint-bilateral filters to transform the blurry response maps from multiple color-line poselet detectors to yield edge-aligned shape cues. It leads to promising human identification and co-segmentation performance.

4. Fully-supervised methods

Fully-Supervised methods train a segmentation model by using fully annotated data or labeled data in very fine-grained form (e.g. all pixels in a train image are annotated). Then the segmentation model is used to rank or segment unseen data. Supervised methods achieve state-of-the-art performance for certain tasks, such as scene parsing. However, fully-supervised methods require expensive labeling cost, and are only applicable to limited object classes or tasks.

4.1. Object proposals

Automatically and precisely segmenting out objects from an image is still an unsolved problem. Instead of searching for precise object segmentation in one shot, object proposal methods seek to generate a pool of regions that have high probability to cover the objects. This type of methods are based on the assumption that there are general rules to separate object regions from the "stuff", where an "object" tends to have clear size and shape (e.g. pedestrian, car) and highly contrasted with background, as opposed to "stuff"(e.g. sky or grass) which tends to be homogeneous or with recurring patterns of fine-scale structure. Existing object proposal methods can be classified into two groups: *class-specific methods*

Fig. 6. Given a user's photo stream about certain event, which consists of finite objects, e.g. red-girl, blue-girl, blue-baby and apple basket. Each image contains an unknown subset of them, which we called "*Multiple Foreground Cosegmentation*" problem. Kim and Xing [22] proposed the first method to extract irregularly occurred objects from the photo stream (b) by using a few bounding boxes provided by the user in (a). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and *class-agnostic methods*, according to whether class specific information has been introduced.

Class-specific object proposal methods: produce the object region pools by incorporating the class-specific object detectors. The object detectors can be any bounding box detector, e.g. the Deformable Part Model (DPM) [31] or Poselets [115]. For example, Larlus and Jurie [116] obtained the object pools by refining the bounding box using graph cut. Gu et al. [45] proposed to use hierarchical regions for object detection, instead of bounding boxes. However, class-specific object segmentation methods can only be applied to a limited number of object classes.

Class-agnostic object proposals: inspired by the objectness window of Alexe et al. [117], class-independent region proposals directly attempt to produce general object region pool. The underlying rationale for class-agnostic proposals to work is that the object-of-interest is typically distinct from background in certain appearance or geometry cues. Hence, in some sense, class-agnostic object proposal problem is correlated with the popular salient object detection problem. A more comprehensive survey of state-of-the-art objectness window methods and salient object detection methods can be found in [118–120], respectively. Here we focus on discussing the region proposal methods.

One group of class-independent object proposal methods is to generate region proposals by borrowing ideas from interactive methods using low-level cues (e.g. color, texture, shape, etc.), and then a pre-trained classifier is used to rank proposals according to various region features. The representative works include CPMC [46] and Category Independent Object Proposal [47], which extend GrabCut to this scenario by using different seeding techniques (see Fig. 7 for an example). In particular, CPMC applies regularly placed grids as foreground seeds and the image frame boundary as background seeds to train the Gaussian mixture models, then a sequential GraphCut is used to generate a set of foreground region pools, while Category Independent Object Proposal samples foreground seeds from occlusion boundaries inferred by [121]. Then the generated regions proposals are fed to random forests classifier or regressor, which are trained with the object region features (such as graph properties, color contrast, and shape) of the ground truth object regions, for re-ranking. One bottleneck of such object proposal methods is that re-computing GrabCut and appearance model implicitly re-computing some distances, which makes such methods very slow (nearly 2 min/image). A recent work [122] which re-uses the graph cuts over different seeds to accelerate the speed. Instead of computing expensive GrabCut, another recent work called Geodesic Object Proposal (GOP) [123] proposed to place seeds by using RankSVM and exploits the fast geodesic distance transform with dynamic programming for object proposals.

Another way to generate region proposals is to use the region merging method described in Section 2. Then, the region hierarchies are filtered using the classifiers of CPMC. This pipeline has been used in recent MCG [124]. MCG first use the region hierarchies generated by gPb-OWT-UCM, then the regions up to four regions from each scale are merged together to form region candidates. As it is derived from high accuracy contour map, such method can achieve better results than GrabCut based methods. Although such method is limited by the performance of contour detectors, its shortcomings on speed and accuracy have been greatly improved by some recently introduced learning based methods such as Structured Forest [125] or the multi-resolution eigen-solvers [124].

Different from the afore-mentioned methods, which use single strategy to generate object regions, another group of methods apply diversified strategies to produce region proposals, which we call *diversified region proposal* methods. This type of methods typically just produce diversified region proposals, but do not train classifier to rank the proposals. For example, SelectiveSearch [126] generates region trees from superpixels to capture objects at multiple scales by using different merging similarity metrics (such as RGB, Intensity, and Texture). To increase the degree of diversification, SeleciveSearch also applies different parameters to generate initial atomic superpixels. After different segmentation trees are generated, the detection starts from the regions in higher hierarchies. Manen et al. [127] proposed a similar method which exploits merging randomized trees with learned similarity metrics.

4.2. Semantic image parsing

Semantic image parsing aims to break an image into non-overlapped regions which correspond to predefined semantic classes (e.g. car, grass, sheep, etc), as shown in Fig. 8. The popularity of semantic parsing since early 2000s is deeply rooted in the success of some specific computer vision tasks such as face/object detection and tracking, camera pose estimation, multiple view 3D reconstruction and fast conditional random field solvers. The ultimate goal of semantic parsing is to equip the computer with the holistic ability to understand the visual world around us. Although also depending on the given information, high-level learned representations make it different from the interactive methods. This type of methods are also different from the object region proposal methods in the sense that it aims to parse an image as a whole into the "thing" and "stuff" classes, instead of just producing possible "thing" candidates.

Most state-of-the-art image parsing systems are formulated as the problem of finding the most probable labeling on a Markov

Fig. 7. An example of class-independent object proposal [46].

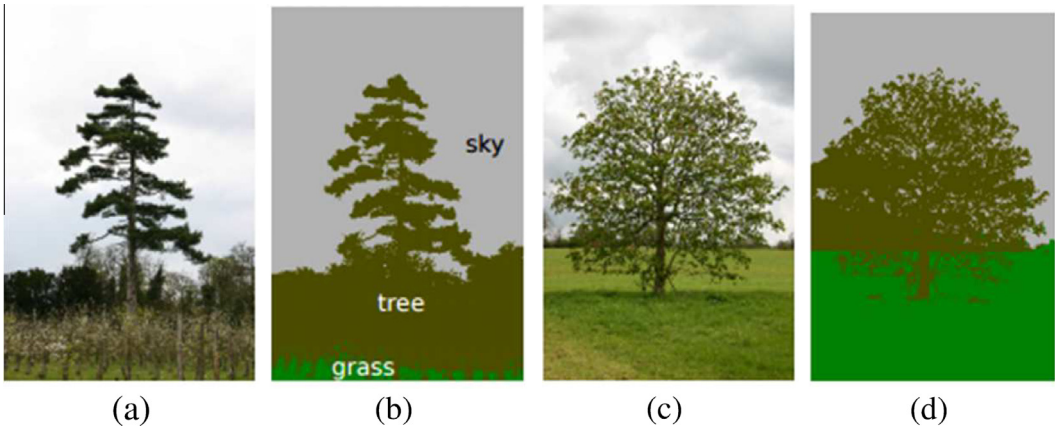

Fig. 8. An example of semantic segmentation [38] which is trained using pictures with human labeled ground truth such as (b) to segment the test image in (c) and produce the final segmentation in (d).

random field (MRF) or conditional random field (CRF). CRF provides a principled probabilistic framework to model complex interactions between output variables and observed features. Thanks to the ability to factorize the probability distribution over different labeling of the random variables, CRF allows for compact representations and efficient inference. The CRF model defines a Gibbs distribution of the output labeling $\mathbf{y}$ conditioned on observed features $\mathbf{x}$ via an energy function $E(\mathbf{y};\mathbf{x})$:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} exp\{-E(\mathbf{y};\mathbf{x})\} \tag{10}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ is a vector of random variables y_i (n is the number of pixels or regions) defined on each node i (pixel or superpixel), which takes a label from a predefined label set L given the observed features $\mathbf{x}$. $Z(\mathbf{x})$ here is called partition function which ensures the distribution is properly normalized and summed to one. Computing the partition function is intractable due to the sum of exponential functions. On the other hand, such computation is not necessary given the task is to infer the most likely labeling. Maximizing a posterior of (10) is equivalent to minimize the energy function $E(\mathbf{y};\mathbf{x})$. A common model for pixel labeling involves a unary potential $\psi^u(y_i;x)$ which is associated with each pixel, and a pairwise potential $\psi^p(y_i, y_j)$ which is associated with a pair of neighborhood pixels:

$$E(\mathbf{y};\mathbf{x}) = \sum_{i=1}^{n} \psi^u(y_i;\mathbf{x}) + \sum_{v_i, v_j \in \mathcal{N}} \psi^p_{ij}(y_i, y_j;\mathbf{x}) \tag{11}$$

Given the energy function, semantic image parsing usually follows the pipelines: (1) Extract features from a patch centered on each pixel; (2) With the extracted features and the ground truth labels, an appearance model is trained to produce a compatible score for each training sample; (3) The trained classifier is applied on the test image's pixel-wise features, and the output is used as the unary term; (4) The pairwise term of the CRF is defined over a 4 or 8-connected neighborhood for each pixel; (5) Perform maximum a posterior (MAP) inference on the graph. Following this common pipeline, there are different variants in different aspects:

- **Features**: The commonly used features are bottom-up pixel-level features such as color or texton. He et al. [128] proposed to incorporate the region and image level features. Shotton et al. [129] proposed to use spatial layout filters to represent the local information corresponding to different classes, which was later adapted to random forest framework for real-time parsing [30]. Recently, deep convolutional neural network learned features [130] have also been used to replace the hand-crafted features, which achieves promising performance.
- **Spatial support**: The spatial support in step 1 can be adapted to superpixels which conform to image internal structures and make feature extraction less susceptible to noise. Also by exploiting superpixels, the complexity of the model is greatly reduces from millions of variables to only hundreds or thousands. Hoiem et al. [131] used multiple segmentations to find out the most feasible configuration. Tighe and Lazebnik [132] used superpixels to retrieve similar superpixels in the training set to generate unary term. To handle multiple superpixel hypotheses, Ladicky et al. [133] proposed the robust higher order potential, which enforces the labeling consistency between the superpixels and their underlying pixels.
- **Context**: The context (such as boat in the water, car on the road) has emerged as another important factor beyond the basic smoothness assumption of the CRF model. Basic context model is implicitly captured by the unary potential, e.g. the pixels with green colors are more likely to be grass class. Recently, more sophisticated class co-occurrence information has been incorporated in the model. Rabinovich et al. [134] learn label co-occurence statistic in the training set and then incorporated it into CRF as additional potential. Later the systems using multiple forms of context based on co-occurence, spatial adjacency and appearance have been proposed in [135,136,132]. Ladicky et al. [39] proposed an efficient method to incorporate global context, which penalizes unlikely pairs of labels to be assigned anywhere in the image by introducing one additional variable in the GraphCut model.
- **Combination of 'Thing' and 'Stuff'**: The combination of high-level and low-level information has also been considered in recent scene parsing works. The low-level information (e.g. color, texture) is better at capturing 'stuff' class (e.g. sky or water) which is homogeneous. On the other hand, the high-level information such as object detectors are good at capturing 'thing' class (e.g. sheep or car). Therefore, their combination helps develop a more holistic scene understanding system. There are some recent studies incorporating the sliding window detectors such as Deformable Part Model [137] or Poselet [115]. Specifically, Ladicky et al. [138] proposed the higher-order robust potential based on detectors which use GrabCut to generate the shape mask to compete with bottom up cues. Floros et al. [139] instead inferred the shape mask from the Implicit Shape Model segmentation system [140]. Arbelaez et al. [141] used the Poselet detector to segment articulated objects. Guo and Hoiem [142] chose to use Auto-Context [143] to

incorporate the detector response in their system. Xia et al. [144] apply CPMC in Section 4.1 on object detection's bounding boxes to generate object proposals and then the proposals are subsequently fused and refined to form the final segmentation. More recently, Tighe and Lazebnik [37] proposed to transfer the mask of training images to test images as the shape potential by using trained exemplar SVM model which greatly improve the performance of the non-parametric scene parsing system.

- **Inference**: To optimize the energy function, various techniques can be used, such as GraphCut, Belief-Propagation or Primal–Dual methods. A complete review of recent inference methods can be found in [145]. Original CRF or MRF models are usually limited to 4-neighbor or 8-neighbor. Recently, the fully connected graphical model which connects all pixels has also become popular due to the availability of efficient approximation of the time–costly message-passing step via fast image filtering [38], with the requirement that the pairwise term should be a mixture of Gaussian kernels. Vineet et al. [146] introduced the higher-order term to the fully connected CRF framework, which generalizes its application.
- **Non-parametric approaches**: The current pipeline needs pre-trained parametric classifier (e.g. RandomForest or SVM), which is quite restrictive when new classes are included in the database. Recently some researchers have advocated for non-parametric, data-driven approaches for open-universe datasets. Such approaches avoid training by retrieving similar training images from the database for segmenting the new image. Liu et al. [36] proposed to use SIFT-flow [147] to transfer masks from train images. On the other hand, Tighe and Lazebnik [132] proposed to retrieve nearest superpixel neighbor in training images and achieved comparable performance to [36].

5. Dataset and evaluation metrics

5.1. Datasets

To inspire new methods and objectively evaluate their performance for certain applications, different datasets and evaluation metrics have been proposed. Initially, the huge labeling cost limits the size of the datasets [148,18] (typically in hundreds of images). Recently, with the popularity of crowdsourcing platform such as Amazon Merchant Turk (AMT) and LabelMe [149], the label cost is shared over the internet users, which makes large datasets with millions of images and labels possible. Below we summarize the most influential datasets which are widely used in the existing segmentation literature ranging from unsupervised image segmentation to fully-supervised scene understanding:

5.1.1. Single image segmentation datasets

Berkeley segmentation benchmark dataset (BSDS) [148] is one of the earliest and largest datasets for contour detection and single image object-agnostic segmentation with human annotation. The latest BSDS dataset contains 200 images for training, 100 images for validation and the rest 200 images for testing. Each image is annotated by at least 3 subjects. Though the size of the dataset is small, it still remains one of the most difficult segmentation datasets as it contains various object classes with great pose variation, background clutter and other challenges. It has also been used to evaluate superpixel segmentation methods. Recently, Li et al. [150] proposed a new benchmark based on BSDS, which can evaluate semantic segmentation at object or part level.

MSRC-interactive segmentation dataset [18] includes 50 images with a single binary ground-truth for evaluating interactive segmentation accuracy. This dataset also provides imitated inputs such as labeling-lasso and rectangle with labels for background, foreground and unknown areas.

5.1.2. Cosegmentation datasets

MSRC-cosegmentation dataset [94] has been used to evaluate image-pair binary cosegmentation. The dataset contains 25 image pairs with similar foreground objects but heterogeneous backgrounds, which matches the assumptions of early cosegmentation methods [94,95,29]. Some pairs of the images are picked such that they contain some camouflage to balance database bias which forms the baseline cosegmentation dataset.

iCoseg dataset [96] is a large binary-class image cosegmentation dataset for more realistic scenarios. It contains 38 groups with a total of 643 images. The content of the images ranges from wild animals, popular landmarks, sports teams to other groups containing similar foregrounds. Each group contains images of similar object instances from different poses with some variations in the background. iCoseg is challenging because the objects are deformed considerably in terms of viewpoint and illumination, and in some cases, only a part of the object is visible. This contrasts significantly with the restrictive scenario of MSRC-Cosegmentation dataset.

FlickrMFC dataset [22] is the only dataset for multiple foreground cosegmentation, which consists of 14 groups of images with manually labeled ground-truth. Each group includes 10–20 images which are sampled from a Flikcr photostream. The image content covers daily scenarios such as children-playing, fishing, and sports. This dataset is perhaps the most challenging cosegmentation dataset as it contains a number of repeating subjects that are not necessarily presented in every image and there are strong occlusions, lighting variations, or scale or pose changes. Meanwhile, serious background clutters and variations often make even state-of-the-art object detectors failing on these realistic scenarios.

5.1.3. Video segmentation/cosegmentation datasets

SegTrack dataset [151] is a large binary-class video segmentation dataset with pixel-level annotation for primary objects. It contains six videos (bird, bird-fall, girl, monkey-dog, parachute and penguin). The dataset contains challenging cases including foreground/background occlusion, large shape deformation and camera motion.

CVC binary-class video cosegmentation dataset [152] contains 4 synthesis videos which paste the same foreground to different backgrounds and 2 videos sampled from the SegTrack. It forms a restrictive dataset for early video cosegmentation methods.

MPI multi-class video cosegmentation dataset [153] was proposed to evaluate video cosegmentation approaches in more challenging scenarios, which contain multi-class objects. This challenging dataset contains 4 different video sets sampled from Youtube including 11 videos with around 520 frames with ground truth. Each video set has different numbers of object classes appearing in them. Moreover, the dataset includes challenging lighting, motion blur and image condition variations.

Cambridge-driving video dataset (CamVid) [154] is a collection of videos with labels of 32 semantic classes (e.g. building, tree, side-walk, traffic light, etc), which are captured by a position-fixed CCTV-camera on a driving automobile over 10 min at 30 Hz footage. This dataset contains 4 video sequences, with more than 700 images at resolution of 960×720. Three of them are sampled at the day light condition and the remaining one is sampled at the dark. The number and the heterogeneity of the object classes in each video sequence are diverse.

5.1.4. Static scene parsing datasets

MSRC 23-class dataset [30] consists of 23 classes and 591 images. Due to the rarity of 'horse' and 'mountain' classes, these two classes are often ignored for training and evaluation. The remaining 21 classes contain diverse objects. The annotated ground-truth is quite rough.

PASCAL VOC dataset [155] provides a large-scale dataset for evaluating object detection and semantic segmentation. Starting from the initial 4-class objects in 2005, now PASCAL dataset includes 20 classes of objects under four major categories (animal, person, vehicle and indoor). The latest train/val dataset has 11,530 images and 6929 segmentations.

LabelMe + SUN dataset: LabelMe [149] is initiated by the MIT CSCAIL which provides a dataset of annotated images. The dataset is still growing. It contains copyright-free images and is open to public contribution. As of October 31, 2010, LabelMe has 187,240 images, 62,197 annotated images, and 658,992 labeled objects. SUN [156] is a sub-sampled dataset from LabelMe. It contains 45,676 image (21,182 indoor and 24,494 outdoor), total 515 object categories. One noteworthy point is that the number of objects in each class is uneven, which can cause unsatisfactory segmentation accuracy for rare classes.

SIFT-flow dataset: The popularity of non-parametric scene parsing requires a large labeled dataset. The SIFT Flow dataset [147] is composed of 2688 images that have been thoroughly labeled by LabelMe users. Liu et al. [147] have split this dataset into 2488 training images and 200 test images and used synonym correction to obtain 33 semantic labels.

Stanford background dataset [35] consists of around 720 images sampled from the existing datasets such as LabelMe, MSRC and PASCAL VOC, whose content consists of rural, urban and harbor scenes. Each image pixel is given two labels: one for its semantic class (sky, tree, road, grass, water, building, mountain and foreground) and the one for geometric property (sky, vertical, and horizontal).

NYU dataset [157]: The NYU-depth V2 dataset is comprised of video sequences from a variety of indoor scenes recorded by both the RGB and depth cameras of Microsoft Kinect. It features 1449 densely labeled pairs of aligned RGB and depth images, 464 new scenes taken from 3 cities, and 407,024 new unlabeled frames. Each object is labeled with a class and an instance number (cup1, cup2, cup3, etc.).

Microsoft COCO [158] is a recent dataset for holistic scene understanding, which provides 328K images with 91 object classes. One substantial difference with other large datasets, such as PASCAL VOC and SUN datasets, is that Microsoft COCO contains more labeled instances in million units. The authors argue that it can facilitate training object detectors with better localization, and learning contextual information.

5.2. Evaluation metrics

As segmentation is an ill-defined problem, how to evaluate an algorithm's goodness remains an open question. In the past, the evaluation was mainly conducted through subjective human inspections or by evaluating the performance of subsequent vision system which uses image segmentation. To objectively evaluate a method, it is desirable to associate the segmentation with perceptual grouping. Current trend is to develop a benchmark [148] which consists of human-labeled segmentation and then compares the algorithm's output with the human-labeled results using some metrics to measure the segmentation quality. Various evaluation metrics have been proposed:

- **Boundary matching**: This method works by matching the algorithm-generated boundaries with human-labeled boundaries, and then computing some metric to evaluate the matching quality. Precision and recall framework proposed by Martin [159] is among the widely used evaluation metrics, where the *Precision* measures the proportion of how many machine-generated boundaries can be found in human-labeled boundaries and is sensitive to over-segmentation, while the *Recall* measures the proportion of how many human-labeled boundaries can be found in machine-generated boundaries and is sensitive to under-segmentation. In general, this method is sensitive to the granularity of human labeling.
- **Region covering**: This method [159] operates by checking the overlap between the machine-generated regions and human-labeled regions. Let S_{seg} and S_{hum} denote the machine segmentation and the human segmentation, respectively. Denote the corresponding segment regions for pixel p_i from the pixel set $P = \{p_1, p_2, \ldots, p_n\}$ as $C(S_{seg}, p_i)$ and $C(S_{hum}, p_i)$. The relative region covering error at p_i is

$$O(S_{seg}, S_{hum}, p_i) = \frac{|C(S_{seg}, p_i) \setminus C(S_{hum}, p_i)|}{C(S_{seg}, p_i)} \tag{12}$$

where $\setminus$ is the set differencing operator.
The global region covering error is defined as:

$$GCE(S_{seg}, S_{hum}) = \frac{1}{n} min\left\{\sum_i O(S_{seg}, S_{hum}, p_i), O(S_{label}, S_{hum}, p_i)\right\} \tag{13}$$

However, when each pixel is a segment or the whole image is a segment, the *GCE* becomes zero which is undesirable. To alleviate these problems, the authors proposed to replace *min* operation by using *max* operation, but such change will not encourage segmentation at finer detail.
Another commonly used region based criterion is the Intersection-over-Union, by checking the overlap between the S_{seg} and S_{hum}:

$$IoU(S_{seg}, S_{hum}) = \frac{S_{seg} \cap S_{hum}}{S_{seg} \cup S_{hum}} \tag{14}$$

- **Variation of Information** (VI): This metric [160] measures the distance between two segmentations S_{seg} and S_{hum} using average conditional entropy. Assume that S_{seg} and S_{hum} have clusters $C_{seg} = \{C^1_{seg}, C^2_{seg}, \ldots, C^K_{seg}\}$ and $C_{hum} = \{C^1_{hum}, C^2_{hum}, \ldots, C^{K'}_{hum}\}$, respectively.
  The variation of information is defined as:

$$VI(S_{seg}, S_{hum}) = H(S_{seg}) + H(S_{hum}) - 2I(S_{seg}, S_{hum}) \tag{15}$$

where $H(S_{seg})$ is the entropy associated with clustering S_{seg}:

$$H(S_{seg}) = -\sum_{k=1}^{K} P(k) log P(k) \tag{16}$$

with $P(k) = \frac{n_k}{n}$, where n_k is the number of elements in cluster k and n is the total number of elements in S_{seg}. $I(S_{seg}, S_{hum})$ is the mutual information between S_{seg} and S_{hum}:

$$I(S_{seg}, S_{hum}) = \sum_{k=1}^{K}\sum_{k'=1}^{K'} P(k,k') log \frac{P(k,k')}{P(k)P(k')} \tag{17}$$

with $P(k,k') = \frac{|C^k_{seg} \bigcap C^{k'}_{hum}|}{n}$.
Although *VI* posses some interesting property, its perceptual meaning and potential in evaluating more than one ground-truth are unknown [55].
- **Probabilistic Random Index** (PRI): PRI was introduced to measure the compatibility of assignments between pairs of elements in S_{seg} and $S_{hum} = \{S^1_{hum}, S^2_{hum}, \ldots, S^n_{hum}\}$. It has been defined to deal with multiple ground-truth segmentations [161]:

$$PRI(S_{seg}, S^k_{hum}) = \frac{1}{\frac{N}{2}} \sum_{i<j} [c_{ij} p_{ij} + (1 - c_{ij})(1 - p_{ij})] \quad (18)$$

where S^k_{hum} is the kth human-labeled ground-truth, c_{ij} is the event that pixels i and j have the same label and p_{ij} is the corresponding probability. As reported in [50], the PRI has a small dynamic range and the values across images and algorithms are often similar which makes the differentiation difficult.

6. Discussions and future directions

6.1. Design choices

When designing a new segmentation algorithm, it is often difficult to make choices among various design choices, e.g. to use superpixel or not, to use more images or not. It all depends on the applications. Our literature review has cast some lights on the pros and cons of some commonly used design choices, which is worth thinking twice before going ahead for a specific setup.

Patch vs. region vs. object proposal: Careful readers might notice that there has been a significant trend in migrating from patch based analysis to region based (or superpixel based) analysis. The continuous performance improvement in terms of boundary recall and execution time makes superpixel a fast preprocessing technique. The advantages of using superpixel lie in not only the time reduction in training and inference but also more complex and discriminative features that can be exploited. On the other hand, superpixel itself is not perfect, which could introduce new structure errors. For users who care more about visual results on the segment boundary, pixel-based or hybrid approach of combining pixel and superpixel should be considered as better options. The structure errors of using superpixel can also be alleviated by using different methods and parameters to produce multiple over-segmentation or using fast edge-aware filtering to refine the boundary. For users more caring about localization accuracy, the region based way is more preferred due to the various advantages introduced while the boundary loss can be neglected. Another uprising trend that is worth mentioning is the application of the object region proposals [100,101,162]. Due to the larger support provided by object-like regions than over-segmentation or pixels, more complex classifiers and region-level features can be extracted. However, the recall rate of the object proposal is still not satisfactory (around 60%); therefore more careful designs need to be made when accuracy is a major concern.

Intrinsic cues vs. extrinsic cues: Although intrinsic cues (the features and prior knowledge for a single image) still play dominant roles in existing CV applications, extrinsic cues which come from multiple images (such as multi-view images, video sequence, and a super large image dataset of similar products) are attracting more and more attentions. An intuitive answer why extrinsic cues convey more semantics can be interpreted in terms of statistic. When there are many signals available, the signals which repeatedly appear form patterns of interest, while those errors are averaged. Therefore, if there are multiple images containing redundant but diverse information, incorporating extrinsic cues should bring some further improvement. When taking extrinsic cues, the source of information needs to be considered in the algorithm design. More robust constraints such as the cues from multiple-view geometric or spatial–temporal relationships should be exploited first. When working with external information such as a large dataset which contains heterogeneous data, a mechanism that can handle noisy information should be developed.

Hand-crafted features vs. learned features: Hand-crafted features, such as intensity, color, SIFT and Bag-of-Word, have played important roles in computer vision. These simple and training-free features have been applied to many applications, and their effectiveness has been widely examined. However, the generalization capability of these hand-crafted features from one task to another task depends on the heuristic feature design and combination, which can compromise the performance. The development of low-level features has become more and more challenging. On the other hand, learned features from labeled database have recently been demonstrated advantages in some applications, such as scene understanding and object detection. The effectiveness of the learned features comes from the context information captured from longer spatial arrangement and higher order co-occurrence. With labeled data, some structured noise is eliminated which helps highlight the salient structures. However, learned features can only detect patterns for certain tasks. If migrating to other tasks, it needs newly labeled data, which is time consuming and expensive to obtain. Therefore, it is suggested to choose features from the hand-craft features first. If it happens to have labeled data, then using learned features usually boost up the performance.

6.2. Promising future directions

Based on the discussed literature and the recent development in segmentation community, here we suggest some future directions which is worth for exploring:

Beyond single image: Although single image segmentation has played a dominant role in the past decades, the recent trend is to use the internal structures of multiple images to facilitate segmentation. When human perceives a scene, the motion cues and depth cues provide extra information, which should not be neglected. For example, when the image data is from a video, the 3D geometry estimated from the structure-from-motion techniques can be used to help image understanding [163,164]. The depth information captured by commodity depth cameras such as Kinect also benefits the indoor scene understanding [165,166]. Beyond 3D information, the useful information from other 2D images in a large database, either organized (like ImageNet) or un-organized (like product or animal category), or multiple heterogeneous datasets of the same category, can also be exploited [110,111].

Toward multiple instances: Most segmentation methods still aim to produce a most likely solution, e.g. all pixels of the same category are assigned the same label. On the other hand, people are also interested in knowing the information of "how many" of the same category objects are there, which is a limitation of the existing works. Recently, there are some works making efforts towards this direction by combining state-of-the-art detectors with the CRF modeling [138,167,168] which have achieved some progress. In addition, the recently developed dataset, Microsoft COCO [158], which contains many images with labeled instances, can be expected to boost the development in this direction.

Become more holistic: Most existing works consider the image labeling task alone [128,30]. On the other hand, studying a related task can improve the existing scene analysis problem. For example, by combining the object detection with image classification, Yao et al. [169] proposed a more robust and accurate system than the ones which perform single task analysis. Such holistic understanding trend can also be seen from other works, by combining context [134,170], geometry [35,171,131,172], attributes [173–177] or even language [178,179]. It is therefore expected that a more holistic system should lead to better performance than those performing monotone analysis, though at an increasing inference cost.

Go from shallow to deep: Feature has been played an important role in many vision applications, e.g. stereo matching, detection and segmentation. Good features, which are highly discriminative to differentiate 'object' from 'stuff', can help object segmentation significantly. However, most features today are

hand-crafted designed for specific tasks. When applied to other tasks, the hand-crafted features might not generalize well. Recently, learned features using multiple-layer neural network [180] have been applied in many vision tasks, including object classification [181], face detection [182], pose estimation [183], object detection [184] and semantic segmentation for nature and RGBD indoor scenes [130,185] with the help of object proposal [186,187] or even some recent works directly proposed to segment by using an end-to-end deep convolutional networks [188,189], and achieved state-of-the-art performance. Therefore, it is interesting to explore whether such learned features can benefit more complex tasks.

7. Conclusion

In this paper, we have conducted a thorough review of recent development of image segmentation methods, including the unsupervised methods, the weak-/semi-supervised methods and the fully-supervised methods. We have discussed the pros and cons of different methods, and suggested some design choices and future directions which are worthwhile for exploration.

Segmentation as a community has achieved substantial progress in the past decades. Although some technologies such as interactive segmentation have been commercialized in recent Adobe and Microsoft products, there is still a long way to make segmentation a general and reliable tool to be widely applied to practical applications. This is more related to the limitations of existing methods in terms of robustness and efficiency. For example, one can always observe that given images under various degradations (strong illumination change, noise corruption or rain situation), the performance of the existing methods could drop significantly [190]. With the rapid improvement in computing hardware, more effective and robust methods will be developed, where the breakthrough is likely to come from the collaborations with other engineering and science communities, such as physics, neurology and mathematics.

Acknowledgments

We thank anonymous reviewers' constructive comments for improving the manuscript. This research is partially supported by MoE AcRF Tier-1 Grant RG30/11.

References

[1] M. Wertheimer, Laws of organization in perceptual forms, in: Psycologische Forschung, 1929, pp. 301–350.
[2] D.R. Martin, C. Fowlkes, J. Malik, Learning to detect natural image boundaries using local brightness, color, and texture cues, IEEE Trans. Pattern Anal. Mach. Intell. 26 (5) (2004) 530–549.
[3] D.D. Hoffman, M. Singh, Salience of visual parts, Cognition 63 (1) (1997) 29–78.
[4] T. Chen, M. Cheng, P. Tan, A. Shamir, S. Hu, Sketch2Photo: internet image montage, ACM Trans. Graph. 28 (5) (2009) 124:1–124:10.
[5] A.Y.S. Chia, S. Zhuo, R.K. Gupta, Y. Tai, S. Cho, P. Tan, S. Lin, Semantic colorization with internet images, ACM Trans. Graph. 30 (6) (2011) 156.
[6] D. Pfeiffer, U. Franke, Efficient representation of traffic scenes by means of dynamic stixels, in: IEEE Intelligent Vehicles Symposium, 2010, pp. 217–224.
[7] T. Goldstein, X. Bresson, S. Osher, Geometric applications of the split Bregman method: segmentation and surface reconstruction, J. Sci. Comput. 45 (1–3) (2010) 272–293.
[8] D. Hoiem, A.A. Efros, M. Hebert, Automatic photo pop-up, ACM Trans. Graph. 24 (3) (2005) 577–584.
[9] R. Ohlander, K. Price, D.R. Reddy, Picture segmentation using a recursive region splitting method, Comput. Graph. Image Process. 8 (1978) 313–333.
[10] C.R. Brice, C.L. Fennema, Scene analysis using regions, Artif. Intell. 1 (3) (1970) 205–226.
[11] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell. 24 (5) (2002) 603–619.
[12] P.F. Felzenszwalb, D.P. Huttenlocher, Efficient graph-based image segmentation, Int. J. Comput. Vision 59 (2) (2004) 167–181.
[13] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.
[14] L. Vincent, P. Soille, Watersheds in digital spaces: an efficient algorithm based on immersion simulations, IEEE Trans. Pattern Anal. Mach. Intell. 13 (6) (1991) 583–598.
[15] R. Beare, A locally constrained watershed transform, IEEE Trans. Pattern Anal. Mach. Intell. 28 (7) (2006) 1063–1074.
[16] T.F. Chan, L.A. Vese, Active contours without edges, IEEE Trans. Image Process. 10 (2) (2001) 266–277.
[17] S. Osher, J.A. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1) (1988) 12–49.
[18] C. Rother, V. Kolmogorov, A. Blake, GrabCut: interactive foreground extraction using iterated graph cuts, ACM Trans. Graph. 23 (3) (2004) 309–314.
[19] W. Yang, J. Cai, J. Zheng, J. Luo, User-friendly interactive image segmentation through unified combinatorial user inputs, IEEE Trans. Image Process. 19 (9) (2010) 2470–2479.
[20] T.N.A. Nguyen, J. Cai, J. Zhang, J. Zheng, Robust interactive image segmentation using convex active contours, IEEE Trans. Image Process. 21 (8) (2012) 3734–3743.
[21] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, F.-F. Li, ImageNet: a large-scale hierarchical image database, in: CVPR, 2009.
[22] G. Kim, E.P. Xing, On multiple foreground cosegmentation, in: CVPR, 2012.
[23] G. Kim, E.P. Xing, F.-F. Li, T. Kanade, Distributed cosegmentation via submodular optimization on anisotropic diffusion, in: ICCV, 2011.
[24] A. Joulin, F. Bach, J. Ponce, Multi-class cosegmentation, in: CVPR, 2012.
[25] J.C. Rubio, J. Serrat, A.M. López, N. Paragios, Unsupervised co-segmentation through region matching, in: CVPR, 2012.
[26] A. Joulin, F.R. Bach, J. Ponce, Discriminative clustering for image co-segmentation, in: CVPR, 2010.
[27] K.-Y. Chang, T.-L. Liu, S.-H. Lai, From co-saliency to co-segmentation: an efficient and fully unsupervised energy minimization model, in: CVPR, 2011.
[28] L. Mukherjee, V. Singh, J. Peng, Scale invariant cosegmentation for image groups, in: CVPR, 2011.
[29] L. Mukherjee, V. Singh, C.R. Dyer, Half-integrality based algorithms for cosegmentation of images, in: CVPR, 2009.
[30] J. Shotton, J.M. Winn, C. Rother, A. Criminisi, TextonBoost for image understanding: multi-class object recognition and segmentation by jointly modeling texture, layout, and context, Int. J. Comput. Vision 81 (1) (2009) 2–23.
[31] P.F. Felzenszwalb, R.B. Girshick, D.A. McAllester, Cascade object detection with deformable part models, in: CVPR, 2010.
[32] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Süsstrunk, SLIC superpixels compared to state-of-the-art superpixel methods, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2274–2282.
[33] D. Hoiem, A.A. Efros, M. Hebert, Putting objects in perspective, Int. J. Comput. Vision 80 (1) (2008) 3–15.
[34] P. Kohli, L. Ladicky, P.H.S. Torr, Robust higher order potentials for enforcing label consistency, Int. J. Comput. Vision 82 (3) (2009) 302–324.
[35] S. Gould, R. Fulton, D. Koller, Decomposing a scene into geometric and semantically consistent regions, in: ICCV, 2009.
[36] C. Liu, J. Yuen, A. Torralba, Nonparametric scene parsing via label transfer, IEEE Trans. Pattern Anal. Mach. Intell. 33 (12) (2011) 2368–2382.
[37] J. Tighe, S. Lazebnik, Finding things: image parsing with regions and per-exemplar detectors, in: CVPR, 2013.
[38] P. Krähenbühl, V. Koltun, Efficient inference in fully connected CRFs with gaussian edge potentials, in: NIPS, 2011.
[39] L. Ladicky, C. Russell, P. Kohli, P.H.S. Torr, Graph cut based inference with co-occurrence statistics, in: ECCV, 2010.
[40] C. Carson, S. Belongie, H. Greenspan, J. Malik, Blobworld: image segmentation using expectation-maximization and its application to image querying, IEEE Trans. Pattern Anal. Mach. Intell. 24 (8) (2002) 1026–1038.
[41] T. Blaschke, Object based image analysis for remote sensing, {ISPRS} J. Photogramm. Rem. Sens. 65 (1) (2010) 2–16.
[42] J. Lu, H. Yang, D. Min, M.N. Do, Patch match filter: efficient edge-aware filtering meets randomized search for fast correspondence field estimation, in: CVPR, 2013.
[43] P. Kohli, M.P. Kumar, P.H.S. Torr, P & beyond: move making algorithms for solving higher order functions, IEEE Trans. Pattern Anal. Mach. Intell. 31 (9) (2009) 1645–1656.
[44] G. Mori, Guiding model search using segmentation, in: ICCV, 2005.
[45] C. Gu, J.J. Lim, P. Arbelaez, J. Malik, Recognition using regions, in: CVPR, 2009.
[46] J. Carreira, C. Sminchisescu, CPMC: automatic object segmentation using constrained parametric min-cuts, IEEE Trans. Pattern Anal. Mach. Intell. 34 (7) (2012) 1312–1328.
[47] I. Endres, D. Hoiem, Category-independent object proposals with diverse ranking, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2) (2014) 222–234.
[48] S. Rao, H. Mobahi, A.Y. Yang, S. Sastry, Y. Ma, Natural image segmentation with adaptive texture and boundary encoding, in: ACCV, 2009.
[49] A.K. Jain, A.P. Topchy, M.H.C. Law, J.M. Buhmann, Landscape of clustering algorithms, in: ICPR, 2004.
[50] P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011) 898–916.

[51] R. Szeliski, Computer Vision – Algorithms and Applications, Texts in Computer Science, Springer, 2011.
[52] J. Shi, J. Malik, Normalized cuts and image segmentation, in: CVPR, 1997.
[53] A.Y. Ng, M.I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: NIPS, 2001.
[54] T. Cour, F. Bénézit, J. Shi, Spectral segmentation with multiscale graph decomposition, in: CVPR, 2005.
[55] P. Arbelaez, M. Maire, C.C. Fowlkes, J. Malik, From contours to regions: an empirical evaluation, in: CVPR, 2009.
[56] X. Bresson, S. Esedoglu, P. Vandergheynst, J.-P. Thiran, S. Osher, Fast global minimization of the active contour/snake model, J. Math. Imag. Vision 28 (2) (2007) 151–167.
[57] T. Pock, A. Chambolle, D. Cremers, H. Bischof, A convex relaxation approach for computing minimal partitions, in: CVPR, 2009.
[58] J. Yuan, E. Bae, X.-C. Tai, Y. Boykov, A continuous max-flow approach to potts model, in: ECCV, 2010.
[59] L. Ambrosio, V. Tortorelli, Approximation of functionals depending on jumps by elliptic functionals via Γ-convergence, Commun. Pure Appl. Math. XLIII (1990) 999–1036.
[60] A. Braides, G. Dal Maso, Non-local approximation of the Mumford–Shah functional, Calc. Var. Partial. Differ. Equ. 5 (4) (1997) 293–322.
[61] A. Braides, Approximation of Free-Discontinuity Problems, Lecture Notes in Mathematics, vol. 1694, Springer, 1998.
[62] A. Chambolle, Image segmentation by variational methods: Mumford and Shah functional and the discrete approximations, SIAM J. Appl. Math. 55 (3) (1995) 827–863.
[63] L.A. Vese, T.F. Chan, A multiphase level set framework for image segmentation using the Mumford and Shah model, Int. J. Comput. Vision 50 (3) (2002) 271–293.
[64] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, Int. J. Comput. Vision 1 (4) (1988) 321–331.
[65] X. Ren, J. Malik, Learning a classification model for segmentation, in: ICCV, 2003.
[66] A. Levinshtein, A. Stere, K.N. Kutulakos, D.J. Fleet, S.J. Dickinson, K. Siddiqi, Turbopixels: fast superpixels using geometric flows, IEEE Trans. Pattern Anal. Mach. Intell. 31 (12) (2009) 2290–2297.
[67] P. Wang, G. Zeng, R. Gan, J. Wang, H. Zha, Structure-sensitive superpixels via geodesic distance, Int. J. Comput. Vision 103 (1) (2013) 1–21.
[68] O. Veksler, Y. Boykov, P. Mehrani, Superpixels and supervoxels in an energy optimization framework, in: ECCV, 2010, pp. 211–224.
[69] Y. Zhang, R.I. Hartley, J. Mashford, S. Burn, Superpixels via pseudo-boolean optimization, in: ICCV, 2011.
[70] M. Liu, O. Tuzel, S. Ramalingam, R. Chellappa, Entropy-rate clustering: cluster analysis via maximizing a submodular function subject to a matroid constraint, IEEE Trans. Pattern Anal. Mach. Intell. 36 (1) (2014) 99–112.
[71] M.V. den Bergh, X. Boix, G. Roig, B. de Capitani, L.J.V. Gool, SEEDS: superpixels extracted via energy-driven sampling, in: ECCV, 2012.
[72] E. Mortensen, B. Morse, W. Barrett, J. Udupa, Adaptive boundary detection using 'live-wire' two-dimensional dynamic programming, in: Computers in Cardiology 1992, Proceedings of, 1992, pp. 635–638.
[73] E.N. Mortensen, W.A. Barrett, Intelligent scissors for image composition, in: SIGGRAPH, 1995, pp. 191–198.
[74] Y. Liu, Y. Yu, Interactive image segmentation based on level sets of probabilities, IEEE Trans. Vis. Comput. Graph. 18 (2) (2012) 202–213.
[75] J. He, C.-S. Kim, C.-C.J. Kuo, Interactive Segmentation Techniques: Algorithms and Performance Evaluation, Springer, 2013.
[76] Y. Boykov, M.-P. Jolly, Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images, in: ICCV, 2001.
[77] L. Grady, Random walks for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 28 (11) (2006) 1768–1783.
[78] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, M. Gelautz, Fast cost-volume filtering for visual correspondence and beyond, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2) (2013) 504–511.
[79] A. Criminisi, T. Sharp, A. Blake, GeoS: geodesic image segmentation, in: ECCV, 2008.
[80] K. McGuinness, N.E. O'Connor, A comparative evaluation of interactive segmentation algorithms, Pattern Recogn. 43 (2) (2010) 434–444.
[81] F. Yi, I. Moon, Image segmentation: a survey of graph-cut methods, in: 2012 International Conference on Systems and Informatics (ICSAI), 2012.
[82] M. Werlberger, T. Pock, M. Unger, H. Bischof, A variational model for interactive shape prior segmentation and real-time tracking, in: SSVM, 2009.
[83] Y. Li, J. Sun, C. Tang, H. Shum, Lazy snapping, ACM Trans. Graph. 23 (3) (2004) 303–308.
[84] P. Kohli, A. Osokin, S. Jegelka, A principled deep random field model for image segmentation, in: CVPR, 2013.
[85] B.L. Price, B.S. Morse, S. Cohen, Geodesic graph cut for interactive image segmentation, in: CVPR, 2010.
[86] X. Bai, G. Sapiro, A geodesic framework for fast interactive image and video segmentation and matting, in: ICCV, 2007.
[87] H. Zhu, J. Zheng, J. Cai, N. Magnenat-Thalmann, Object-level image segmentation using low level cues, IEEE Trans. Image Process. 22 (10) (2013) 4019–4027.
[88] O. Veksler, Star shape prior for graph-cut image segmentation, in: ECCV, 2008.
[89] V. Gulshan, C. Rother, A. Criminisi, A. Blake, A. Zisserman, Geodesic star convexity for interactive image segmentation, in: CVPR, 2010.
[90] S. Vicente, V. Kolmogorov, C. Rother, Graph cut based image segmentation with connectivity priors, in: CVPR, 2008.
[91] F. Durand, J. Dorsey, Fast bilateral filtering for the display of high-dynamic-range images, ACM Trans. Graph. 21 (3) (2002) 257–266.
[92] K. He, J. Sun, X. Tang, Guided image filtering, IEEE Trans. Pattern Anal. Mach. Intell. 35 (6) (2013) 1397–1409.
[93] J. Lu, K. Shi, D. Min, L. Lin, M.N. Do, Cross-based local multipoint filtering, in: CVPR, 2012.
[94] C. Rother, T.P. Minka, A. Blake, V. Kolmogorov, Cosegmentation of image pairs by histogram matching – incorporating a global constraint into MRFs, in: CVPR, 2006.
[95] D.S. Hochbaum, V. Singh, An efficient algorithm for co-segmentation, in: ICCV, 2009.
[96] D. Batra, A. Kowdle, D. Parikh, J. Luo, T. Chen, iCoseg: interactive co-segmentation with intelligent scribble guidance, in: CVPR, 2010.
[97] M.D. Collins, J. Xu, L. Grady, V. Singh, Random walks based multi-image segmentation: quasiconvexity results and GPU-based solutions, in: CVPR, 2012.
[98] F. Meng, H. Li, G. Liu, Image co-segmentation via active contours, in: ISCAS, 2012.
[99] E. Kim, H. Li, X. Huang, A hierarchical image clustering cosegmentation framework, in: CVPR, 2012.
[100] S. Vicente, C. Rother, V. Kolmogorov, Object cosegmentation, in: CVPR, 2011.
[101] F. Meng, H. Li, G. Liu, K.N. Ngan, Object co-segmentation based on shortest path algorithm and saliency model, IEEE Trans. Multimedia 14 (5) (2012) 1429–1441.
[102] J. Sun, J. Ponce, Learning discriminative part detectors for image classification and cosegmentation, in: ICCV, 2013.
[103] J. Dai, Y.N. Wu, J. Zhou, S. Zhu, Cosegmentation and cosketch by unsupervised learning, in: ICCV, 2013.
[104] A. Faktor, M. Irani, Co-segmentation by composition, in: ICCV, 2013.
[105] Z. Wang, R. Liu, Semi-supervised learning for large scale image cosegmentation, in: ICCV, 2013.
[106] H. Zhu, J. Cai, J. Zheng, J. Wu, N. Magnenat-Thalmann, Salient object cutout using Google images, in: ISCAS, 2013.
[107] M. Rubinstein, A. Joulin, J. Kopf, C. Liu, Unsupervised joint object discovery and segmentation in internet images, in: CVPR, 2013.
[108] Y. Chai, V.S. Lempitsky, A. Zisserman, BiCoS: a bi-level co-segmentation method for image classification, in: ICCV, 2011.
[109] Y. Chai, E. Rahtu, V.S. Lempitsky, L.J.V. Gool, A. Zisserman, TriCoS: a tri-level class-discriminative co-segmentation method for image classification, in: ECCV, 2012.
[110] D. Küttel, M. Guillaumin, V. Ferrari, Segmentation propagation in ImageNet, in: ECCV, 2012.
[111] F. Meng, J. Cai, H. Li, On multiple image group cosegmentation, in: ACCV, 2014.
[112] T. Ma, L.J. Latecki, Graph transduction learning with connectivity constraints with application to multiple foreground cosegmentation, in: CVPR, 2013.
[113] H. Zhu, J. Lu, J. Cai, J. Zheng, N. Thalmann, Multiple foreground recognition and cosegmentation: an object-oriented CRF model with robust higher-order potentials, in: WACV, 2014.
[114] H. Zhu, J. Lu, J. Cai, J. Zheng, N. Magnenat-Thalmann, Poselet-based multiple human identification and cosegmentation, in: ICIP, 2014.
[115] L.D. Bourdev, S. Maji, T. Brox, J. Malik, Detecting people using mutually consistent poselet activations, in: ECCV, 2010.
[116] D. Larlus, F. Jurie, Combining appearance models and markov random fields for category level object segmentation, in: CVPR, 2008.
[117] B. Alexe, T. Deselaers, V. Ferrari, Measuring the objectness of image windows, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2189–2202.
[118] J. Hosang, R. Benenson, B. Schiele, How good are detection proposals, really?, in: BMVC, 2014.
[119] H. Zhu, S. Lu, J. Cai, G. Lee, Diagnosing state-of-the-art object proposal methods, in: BMVC, 2015.
[120] A. Borji, M. Cheng, H. Jiang, J. Li, Salient Object Detection: A Survey, CoRR abs. Available from: 1411.5878.
[121] D. Hoiem, A.A. Efros, M. Hebert, Recovering occlusion boundaries from an image, Int. J. Comput. Vision 91 (3) (2011) 328–346.
[122] A. Humayun, F. Li, J.M. Rehg, RIGOR: reusing inference in graph cuts for generating object regions, in: CVPR, 2014.
[123] P. Krähenbühl, V. Koltun, Geodesic object proposals, in: ECCV, 2014.
[124] P.A. Arbeláez, J. Pont-Tuset, J.T. Barron, F. Marqués, J. Malik, Multiscale combinatorial grouping, in: CVPR, 2014.
[125] P. Dollár, C.L. Zitnick, Structured forests for fast edge detection, in: ICCV, 2013.
[126] K.E.A. van de Sande, J.R.R. Uijlings, T. Gevers, A.W.M. Smeulders, Segmentation as selective search for object recognition, in: ICCV, 2011.
[127] S. Manen, M. Guillaumin, L.J.V. Gool, Prime object proposals with randomized prim's algorithm, in: ICCV, 2013.
[128] X. He, R.S. Zemel, M. Carreira-Perpin, Multiscale conditional random fields for image labeling, in: CVPR, 2004.
[129] J. Shotton, M. Johnson, R. Cipolla, Semantic texton forests for image categorization and segmentation, in: CVPR, 2008.
[130] C. Farabet, C. Couprie, L. Najman, Y. LeCun, Learning hierarchical features for scene labeling, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1915–1929.

[131] D. Hoiem, A.A. Efros, M. Hebert, Recovering surface layout from an image, Int. J. Comput. Vision 75 (1) (2007) 151–172.
[132] J. Tighe, S. Lazebnik, Superparsing – scalable nonparametric image parsing with superpixels, Int. J. Comput. Vision 101 (2) (2013) 329–349.
[133] L. Ladicky, C. Russell, P. Kohli, P.H.S. Torr, Associative hierarchical CRFs for object class image segmentation, in: ICCV, 2009.
[134] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, S. Belongie, Objects in context, in: ICCV, 2007.
[135] C. Galleguillos, A. Rabinovich, S. Belongie, Object categorization using co-occurrence, location and appearance, in: CVPR, 2008.
[136] C. Galleguillos, B. McFee, S.J. Belongie, G.R.G. Lanckriet, Multi-class object localization by combining local contextual interactions, in: CVPR, 2010.
[137] P.F. Felzenszwalb, D.A. McAllester, D. Ramanan, A discriminatively trained, multiscale, deformable part model, in: CVPR, 2008.
[138] L. Ladicky, P. Sturgess, K. Alahari, C. Russell, P.H.S. Torr, What, where and how many? Combining object detectors and CRFs, in: ECCV, 2010.
[139] G. Floros, K. Rematas, B. Leibe, Multi-class image labeling with top-down segmentation and generalized robust p_n potentials, in: BMVC, 2011.
[140] B. Leibe, A. Leonardis, B. Schiele, Robust object detection with interleaved categorization and segmentation, Int. J. Comput. Vision 77 (1–3) (2008) 259–289.
[141] P. Arbelaez, B. Hariharan, C. Gu, S. Gupta, L.D. Bourdev, J. Malik, Semantic segmentation using regions and parts, in: CVPR, 2012.
[142] R. Guo, D. Hoiem, Beyond the line of sight: labeling the underlying surfaces, in: A.W. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, C. Schmid (Eds.), ECCV, 2012.
[143] Z. Tu, Auto-context and its application to high-level vision tasks, in: CVPR, 2008.
[144] W. Xia, C. Domokos, J. Dong, L.-F. Cheong, S. Yan, Semantic segmentation without annotating segments, in: ICCV, 2013.
[145] J.H. Kappes, B. Andres, F.A. Hamprecht, C. Schnörr, S. Nowozin, D. Batra, S. Kim, B.X. Kausler, J. Lellmann, N. Komodakis, C. Rother, A comparative study of modern inference techniques for discrete energy minimization problems, in: CVPR, 2013.
[146] V. Vineet, J. Warrell, P.H.S. Torr, Filter-based mean-field inference for random fields with higher-order terms and product label-spaces, Int. J. Comput. Vision 110 (3) (2014) 290–307.
[147] C. Liu, J. Yuen, A. Torralba, Sift flow: dense correspondence across scenes and its applications, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011) 978–994.
[148] D.R. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: ICCV, 2001.
[149] B.C. Russell, A. Torralba, K.P. Murphy, W.T. Freeman, LabelMe: a database and web-based tool for image annotation, Int. J. Comput. Vision 77 (1–3) (2008) 157–173.
[150] H. Li, J. Cai, N.T.N. Anh, J. Zheng, A benchmark for semantic image segmentation, in: ICME, 2013.
[151] D. Tsai, M. Flagg, J.M.Rehg, Motion coherent tracking with multi-label MRF optimization, in: BMVC, 2010.
[152] J.C. Rubio, J. Serrat, A.M. López, Video co-segmentation, in: ACCV, 2012.
[153] W. chen Chiu, M. Fritz, Multi-class video co-segmentation with a generative multi-video model, in: CVPR, 2013.
[154] G.J. Brostow, J. Shotton, J. Fauqueur, R. Cipolla, Segmentation and recognition using structure from motion point clouds, in: ECCV, 2008.
[155] M. Everingham, L.J.V. Gool, C.K.I. Williams, J.M. Winn, A. Zisserman, The pascal visual object classes (VOC) challenge, Int. J. Comput. Vision 88 (2) (2010) 303–338.
[156] J. Xiao, J. Hays, K.A. Ehinger, A. Oliva, A. Torralba, SUN database: large-scale scene recognition from abbey to zoo, in: CVPR, 2010.
[157] P.K. Nathan Silberman, Derek Hoiem, R. Fergus, Indoor segmentation and support inference from RGBD images, in: ECCV, 2012.
[158] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft COCO: common objects in context, in: ECCV, 2014.
[159] D.R. Martin, An Empirical Approach to Grouping and Segmentation, Ph.D. Thesis, EECS Department, University of California, Berkeley, August 2003.
[160] M. Meila, Comparing clusterings by the variation of information, in: COLT, 2003, pp. 173–187.
[161] R. Unnikrishnan, C. Pantofaru, M. Hebert, Toward objective evaluation of image segmentation algorithms, IEEE Trans. Pattern Anal. Mach. Intell. 29 (6) (2007) 929–944.
[162] J. Dong, Q. Chen, S. Yan, A. Yuille, Towards unified object detection and semantic segmentation, in: ECCV, 2014.
[163] P. Sturgess, K. Alahari, L. Ladicky, P.H.S. Torr, Combining appearance and structure from motion features for road scene understanding, in: BMVC, 2009.
[164] C. Zhang, L. Wang, R. Yang, Semantic segmentation of urban scenes using dense depth maps, in: ECCV, 2010.
[165] N. Silberman, R. Fergus, Indoor scene segmentation using a structured light sensor, in: ICCV Workshops, 2011.
[166] N. Silberman, D. Hoiem, P. Kohli, R. Fergus, Indoor segmentation and support inference from RGBD images, in: ECCV, 2012.
[167] X. He, S. Gould, An exemplar-based CRF for multi-instance object segmentation, in: CVPR, 2014.
[168] J. Tighe, M. Niethammer, S. Lazebnik, Scene parsing with object instances and occlusion ordering, in: CVPR, 2014.
[169] J. Yao, S. Fidler, R. Urtasun, Describing the scene as a whole: joint object detection, scene classification and semantic segmentation, in: CVPR, 2012.
[170] S.K. Divvala, D. Hoiem, J. Hays, A.A. Efros, M. Hebert, An empirical study of context in object detection, in: CVPR, 2009.
[171] A. Gupta, S. Satkin, A.A. Efros, M. Hebert, From 3d scene geometry to human workspace, in: CVPR, 2011.
[172] V. Vineet, J. Warrell, P.H.S. Torr, Filter-based mean-field inference for random fields with higher-order terms and product label-spaces, in: ECCV, 2012.
[173] A. Farhadi, I. Endres, D. Hoiem, Attribute-centric recognition for cross-category generalization, in: CVPR, 2010.
[174] N. Kumar, A.C. Berg, P.N. Belhumeur, S.K. Nayar, Describable visual attributes for face verification and image search, IEEE Trans. Pattern Anal. Mach. Intell. 33 (10) (2011) 1962–1977.
[175] C.H. Lampert, H. Nickisch, S. Harmeling, Learning to detect unseen object classes by between-class attribute transfer, in: CVPR, 2009.
[176] J. Tighe, S. Lazebnik, Understanding scenes on many levels, in: ICCV, 2011.
[177] S. Zheng, M. Cheng, J. Warrell, P. Sturgess, V. Vineet, C. Rother, P.H.S. Torr, Dense semantic image segmentation with objects and attributes, in: CVPR, 2014.
[178] A. Gupta, L.S. Davis, Beyond nouns: exploiting prepositions and comparative adjectives for learning visual classifiers, in: D.A. Forsyth, P.H.S. Torr, A. Zisserman (Eds.), ECCV, 2008.
[179] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A.C. Berg, T.L. Berg, Babytalk: understanding and generating simple image descriptions, IEEE Trans. Pattern Anal. Mach. Intell. 35 (12) (2013) 2891–2903.
[180] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: NIPS, 2012.
[181] D. Erhan, C. Szegedy, A. Toshev, D. Anguelov, Scalable Object Detection using Deep Neural Networks, CoRR abs. Available from: 1312.2249.
[182] A. Toshev, C. Szegedy, DeepPose: Human Pose Estimation via Deep Neural Networks, CoRR abs. Available from: 1312.4659.
[183] C. Szegedy, A. Toshev, D. Erhan, Deep neural networks for object detection, in: NIPS, 2013.
[184] R.B. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: CVPR, 2014.
[185] C. Couprie, C. Farabet, L. Najman, Y. LeCun, Indoor semantic segmentation using depth information, in: JMLR, 2014.
[186] B. Hariharan, P.A. Arbeláez, R.B. Girshick, J. Malik, Simultaneous detection and segmentation, in: ECCV, 2014.
[187] J. Dai, K. He, J. Sun, Convolutional feature masking for joint object and stuff segmentation, in: CVPR, 2015.
[188] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, P. Torr, Conditional random fields as recurrent neural networks, in: ICCV, 2015.
[189] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: CVPR, 2015.
[190] W.T. Freeman, Where computer vision needs help from computer science, in: SODA, 2011.