What Makes for Effective Detection Proposals?

Jan Hosang, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele

Abstract—Current top performing object detectors employ detection proposals to guide the search for objects, thereby avoiding exhaustive sliding window search across images. Despite the popularity and widespread use of detection proposals, it is unclear which trade-offs are made when using them during object detection. We provide an in-depth analysis of twelve proposal methods along with four baselines regarding proposal repeatability, ground truth annotation recall on PASCAL, ImageNet, and MS COCO, and their impact on DPM, R-CNN, and Fast R-CNN detection performance. Our analysis shows that for object detection improving proposal localisation accuracy is as important as improving recall. We introduce a novel metric, the average recall (AR), which rewards both high recall and good localisation and correlates surprisingly well with detection performance. Our findings show common strengths and weaknesses of existing methods, and provide insights and metrics for selecting and tuning proposal methods.

Index Terms—Computer vision, object detection, detection proposals

1. 分组 proposal 方法（Grouping proposal methods）

Grouping proposal methods尝试产生对应于目标的多个区域（可能重叠）。根据它们产生 proposal 的方式可以划分为三类：superpixels (SP)，graph cut (GC) 和 edge contours (EC)。

• **SelectiveSearch (SP)** [15], [29]：通过贪婪地合并超像素来产生 proposals。这个方法没有学习的参数，合并超像素的特征和相似函数是手动设定的。它被 R-CNN 和 Fast R-CNN detectors [8], [16] 等最新的目标检测方法选用。
• **RandomizedPrim's (SP)** [26]：使用类似与SelectiveSearch 的特征，但是使用了一个随机的超像素合并过程来学习所有的可能（probabilities）。此外，速度有了极大地提升。
• **Rantalankila (SP)** [27]：使用类似与SelectiveSearch 的策略，但使用了不同的特征。在后续阶段，产生的区域用作求解图切割的种子点（seeds ）（类似于CPMC）。
• **Chang (SP)** [38]：结合 saliency 和 Objectness 在一个图模型中来合并超像素实现前景/背景（figure/background）分割。
• **CPMC (GC)** [13],[19]：避免初始的分割，使用几个不同的种子点（seeds ）和位元（unaries ）对像素直接进行图切割。生成的区域使用一个大的特征池来排序。
• **Endres (GC)** [14], [21]：从遮挡的边界建立一个分层（hierarchical ）的分割，并且使用不同的种子点和参数来切割图产生区域。产生的 使用大量的线索和鼓励多样性的角度排序。
• **Rigor (GC)** [28]：是 CPMC 的一个改进，使用多个图切割和快速的边缘检测子来加快计算速度。
• **Geodesic (EC)** [22]：首先使用 [36] 对图片过分割。分类器用来为一个测地距离变换标定种子点。每个距离转换的水平集（Level sets）定义了（figure/ground）的分割。
• **MCG (EC)** [23]：基于 [36]， 提出一个快速的用于计算多尺度（multi-scale）层次分割进程。使用边缘强度来合并区域，生成的目标假设（object hypotheses ）使用类似于尺度，位置，形状和边缘强度的线索来排序。

K. van de Sande, J. Uijlings, T. Gevers, and A. Smeulders, “Segmentation as selective search for object recognition,” in Proc. IEEE Int. Conf. Comput. Vis., 2011,pp. 1879–1886.

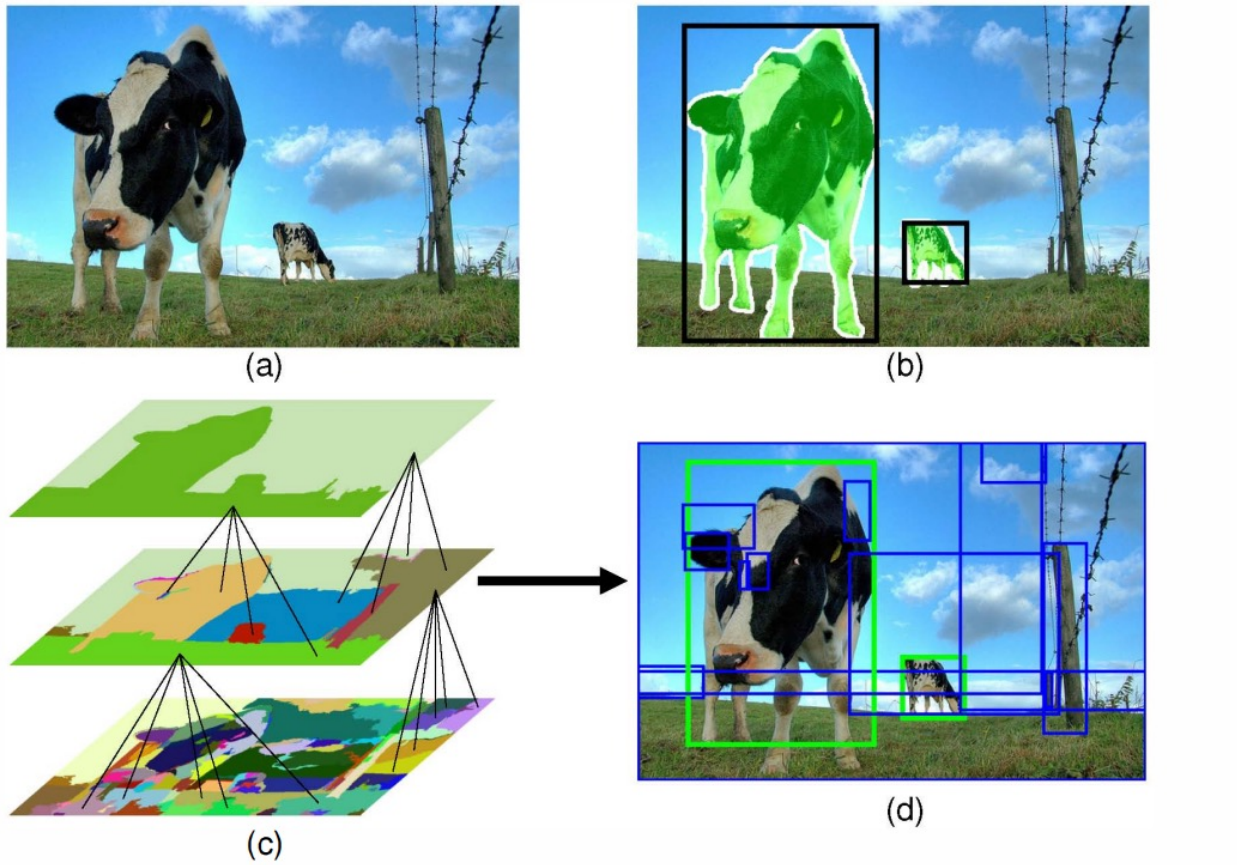
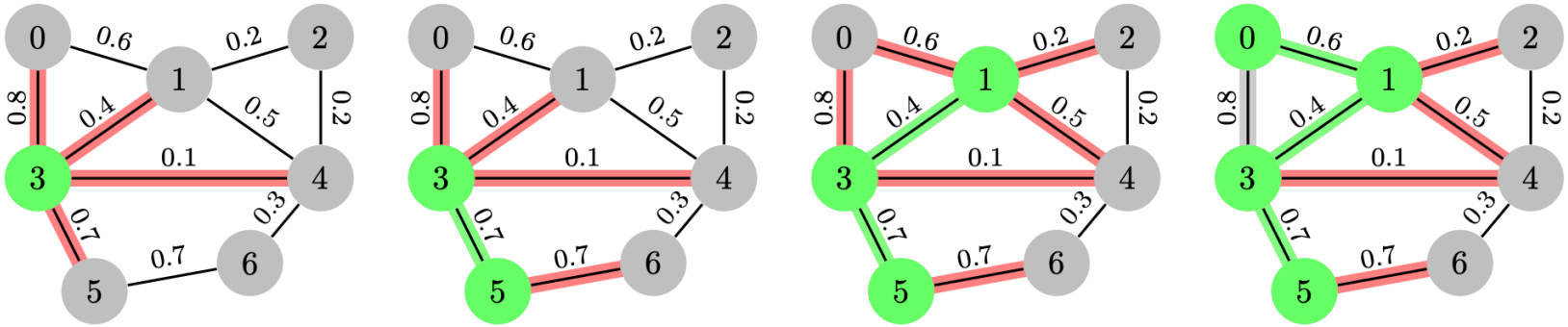


Figure 1. Given an image (a) our aim is to find its objects for which the ground truth is shown in (b). To achieve this, we adapt segmentation as a selective search strategy: We aim for high recall by generating locations at all scales and account for many different scene conditions by employing multiple invariant colour spaces. Example object hypotheses are visualised in (d).

SelectiveSearch (SP) [15], [29]：通过贪婪地合并超像素来产生 proposals。这个方法没有学习的参数，合并超像素的特征和相似函数是手动设定的。它被 R-CNN 和 Fast R-CNN detectors [8], [16] 等最新的目标检测方法选用。

Figure 3. Two examples of our hierarchical grouping algorithm showing the necessity of different scales. On the left we find many objects at different scales. On the right we necessarily find the objects at different scales as the girl is contained by the tv.

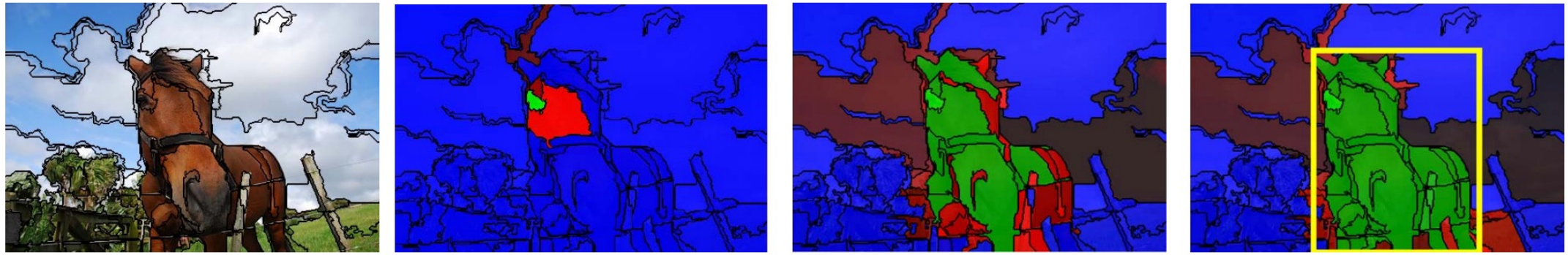
S. Mane n, M. Guillaumin, and L. Van Gool, “Prime object proposals with randomized prim’ s algorithm,” in Proc. IEEE Int. Conf. Comput. Vis., 2013, pp. 2536–2543.



(a) Initialization (b) Iteration 2 (c) Iteration 3 (d) Iteration 4

Figure 2: (a) The Randomized Prim’s algorithm initializes the tree T_1 (green) with a random node. At iteration k (b-d), a new edge is added to the tree T_k. The edges are sampled from the set $\mathcal{E}_k$ (red) of edges connecting T_k to its frontier, proportionally to their edge weights. The Prim’s algorithm corresponds to always selecting the edge in $\mathcal{E}_k$ with maximum weight.

RandomizedPrim's (SP) [26]：使用类似与SelectiveSearch 的特征，但是使用了一个随机的超像素合并过程来学习所有的可能（probabilities）。此外，速度有了极大地提升。



(a) Superpixel segmentation (b) Initialization (c) Iteration k (d) Proposal from partial tree

Figure 3: We apply the Randomized Prim’s algorithm to the connectivity graph of superpixels of an image (a). (b) It starts with one superpixel (green). At each iteration (c), it samples a neighbouring superpixel (red) and decides to add it or return the bounding-box as a proposal (d). The brightness of red indicate the relative probability of sampling superpixels (lighter means more probable). The superpixels in blue are not connected to the current tree, hence cannot be sampled.

J. Carreira and C. Sminchisescu, "Constrained parametric mincuts for automatic object segmentation," in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2010, pp. 3241–3248.

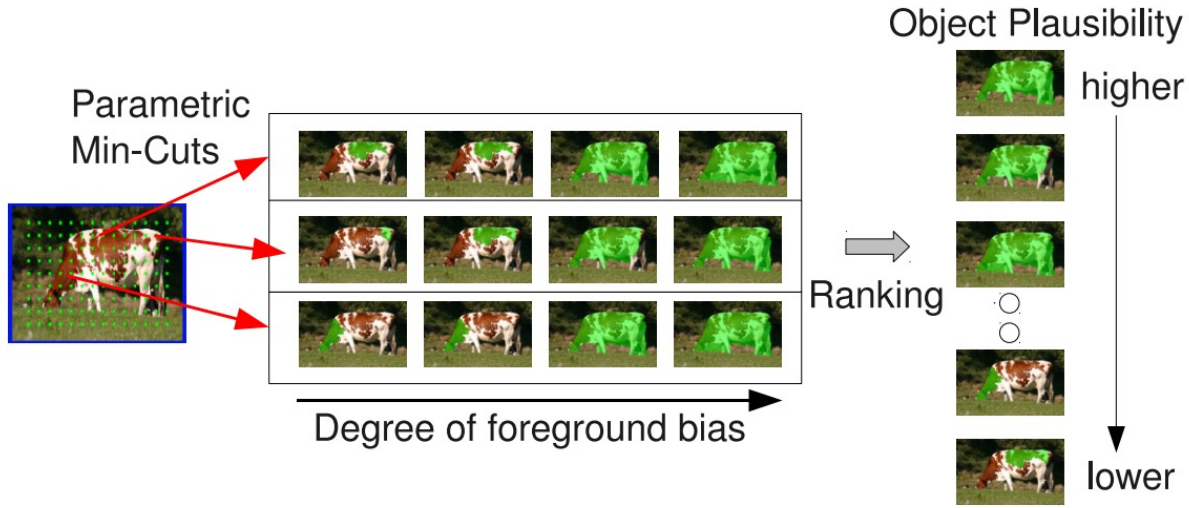


Figure 1. Our object segmentation framework. Segments are extracted around regularly placed foreground seeds, with various background seeds corresponding to image boundary edges, for all levels of foreground bias, which has the effect of producing segments with different scales. The resulting set of segments is filtered and ranked according to their plausibility of being good object hypotheses, based on mid-level properties.

CPMC (GC) [13],[19]：避免初始的分割，使用几个不同的种子点（seeds）和位元（unaries）对像素直接进行图切割。生成的区域使用一个大的特征池来排序。

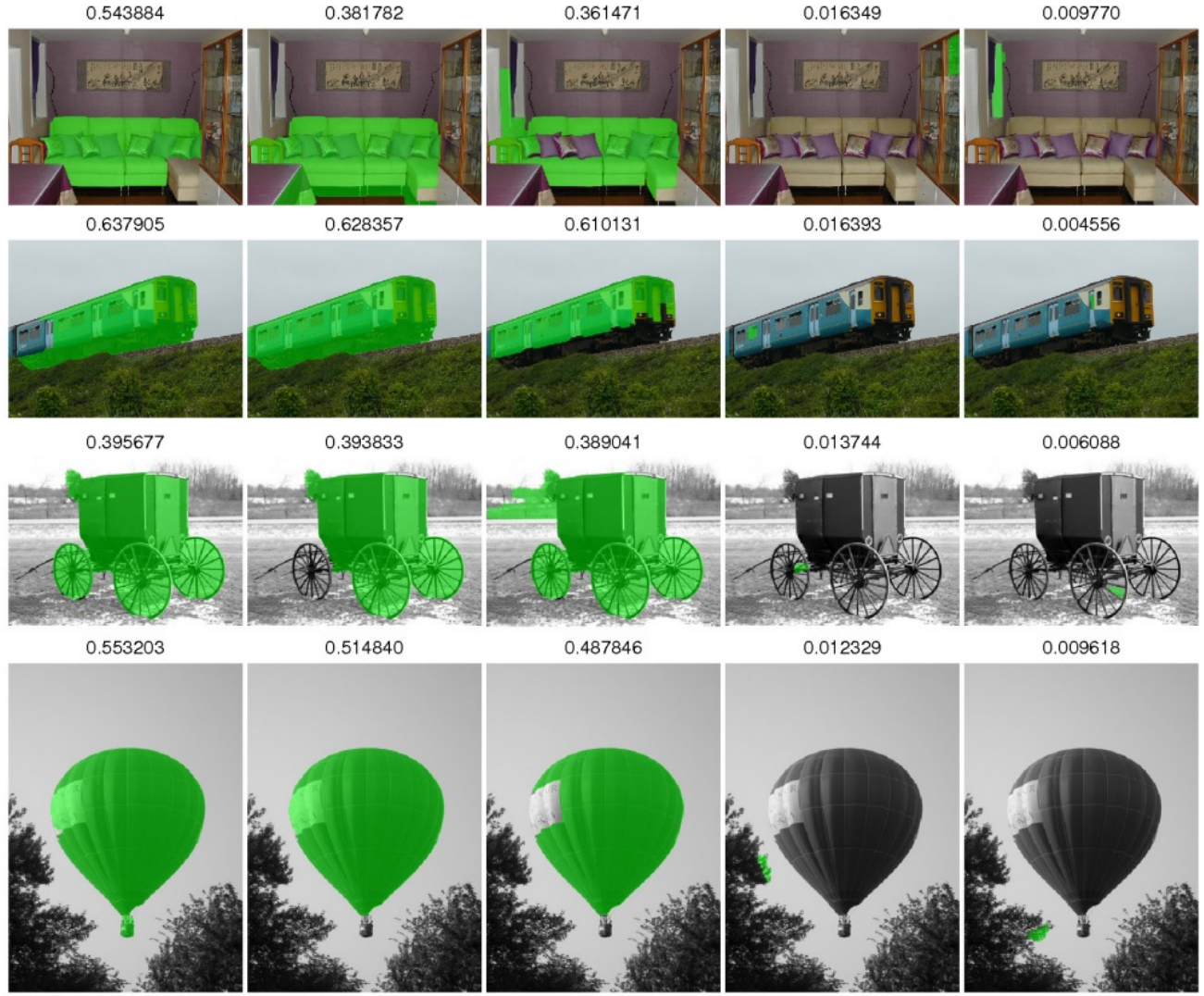


Figure 6. Ranking results obtained using the random forests model learned on the VOC2009 training set. The first three images on each row show the most plausible object hypotheses, the last two images show the least plausible segments. The segment scores are shown above each image. The green regions correspond to the foreground areas of the object hypotheses. The first six rows of images are from the VOC2009 segmentation validation set, the last two are from the Weizmann Segmentation Database. The lowest ranked object hypotheses are usually very small reflecting perhaps the statistics of the images in the VOC2009 training set. The algorithm shows a remarkable preference to segments with large overlap with the objects in the image, although it has no prior knowledge about their classes. There are neither chariots nor balloons in the training set, for example.

A. Humayun, F. Li, and J. M. Rehg, “RIGOR: Recycling inference in graph cuts for generating object regions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2014, pp. 336–343

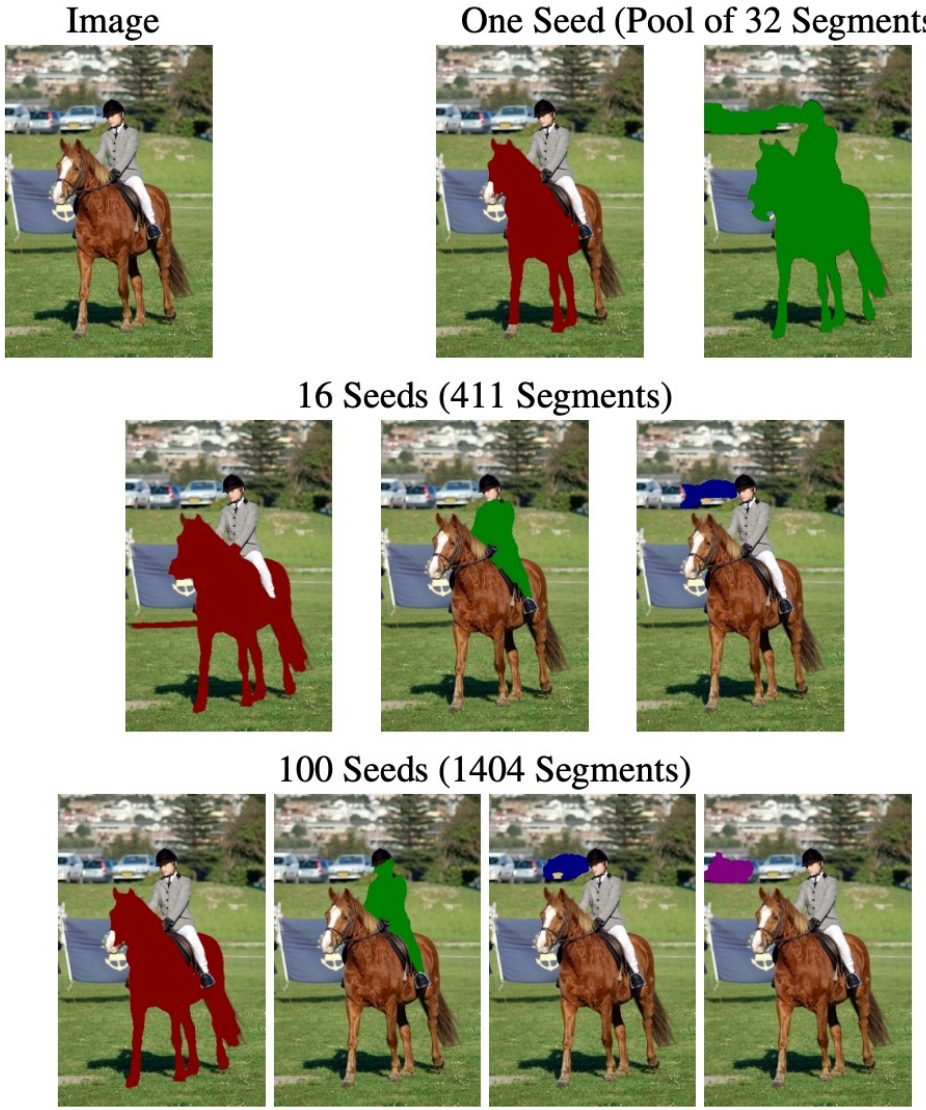
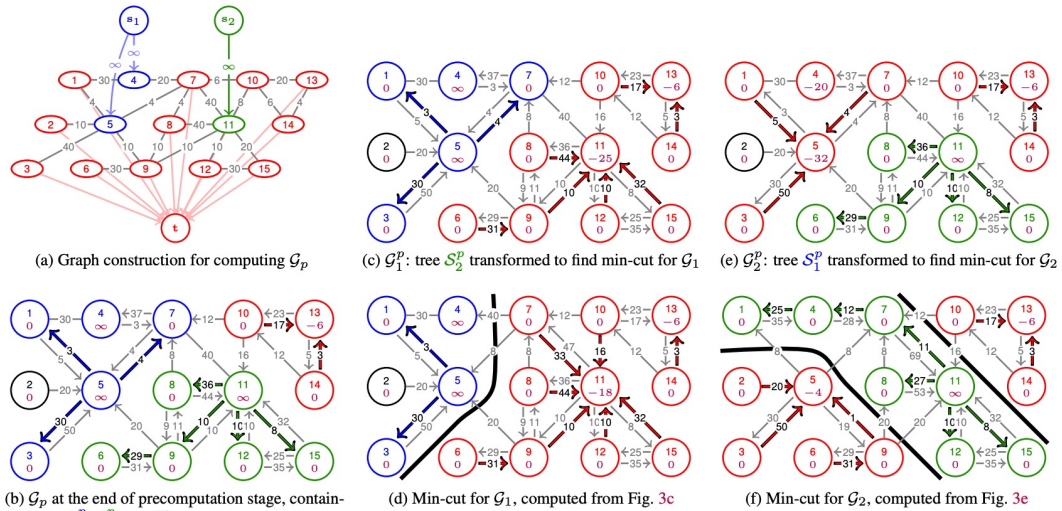


Figure 1: Effect of segment seeds (best viewed in color). With one seed at the center the algorithm is able to segment the horse. With more seeds it finds the person and gradually starts to obtain segments on the cars in the background.

Figure 2: The use of superpixel edges in multiple min-cuts. Given parametric min-cuts from all seed graphs, the color of each edge indicates how many times it was included in a cut (edges separating fg from bg). White indicates an edge that was never used for a cut. A saturated red means the edge was included in many cuts.

Rigor (GC) [28]：是 CPMC 的一个改进，使用多个图切割和快速的边缘检测子来加快计算速度。



(a) Graph construction for computing $\mathcal{G}_p$

(c) $\mathcal{G}_1^p$: tree $\mathcal{S}_2^p$ transformed to find min-cut for $\mathcal{G}_1$

(e) $\mathcal{G}_2^p$: tree $\mathcal{S}_1^p$ transformed to find min-cut for $\mathcal{G}_2$

(b) $\mathcal{G}_p$ at the end of precomputation stage, containing trees $\mathcal{S}_1^p$, $\mathcal{S}_2^p$, and $\mathcal{T}^p$ fully grown

(d) Min-cut for $\mathcal{G}_1$, computed from Fig. 3c

(f) Min-cut for $\mathcal{G}_2$, computed from Fig. 3e

Figure 3: This illustrates min-cut for $\mathcal{G}_1$ and $\mathcal{G}_2$, using the precomputation graph $\mathcal{G}_p$, composed of trees $\mathcal{S}_1^p$, $\mathcal{S}_2^p$ and $\mathcal{T}^p$. The value within each node is the unary residual capacity r_u. The value on each n-edge is its residual capacity r_{uv}. Gray edges, without an arrow, stand-in for bi-directional n-edges. **(a)** shows the precomputation graph where $c_{s4} = c_{s5} = c_{s11} = \infty$, and other nodes are connected to the sink $\mathbf{t}$. The color of the node indicates which tree it belongs to. **(b)** shows the result once the precomputation graph $\mathcal{G}_p$ is built with $\mathcal{S}_1^p$, $\mathcal{S}_2^p$ and $\mathcal{T}^p$. Unsaturated tree edges are given in the color of the tree. Node 2 is free because it cannot be grown into $\mathcal{S}_1^p$. **(c)** shows the transformation of $\mathcal{G}_p$ into graph $\mathcal{G}_1^p$, which is valid for finding a cut for $\mathcal{G}_1$. **(d)** shows the resulting cut after running BK on the graph in (c). Similarly, **(e)** shows the transformed graph $\mathcal{G}_2^p$. **(f)** shows the final cut. Notice that nodes 10, 13, 14 remain in the sink all the time.

P. Arbelaez, J. Pont-Tuset, J. Barron, F. Marque s, and J. Malik, “Multiscale combinatorial grouping,” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2014, pp. 328–335.

MCG (EC) [23]：基于 [36]，提出一个快速的用于计算多尺度（multi-scale）层次分割进程。使用边缘强度来合并区域，生成的目标假设（object hypotheses）使用类似于尺度，位置，形状和边缘强度的线索来排序。

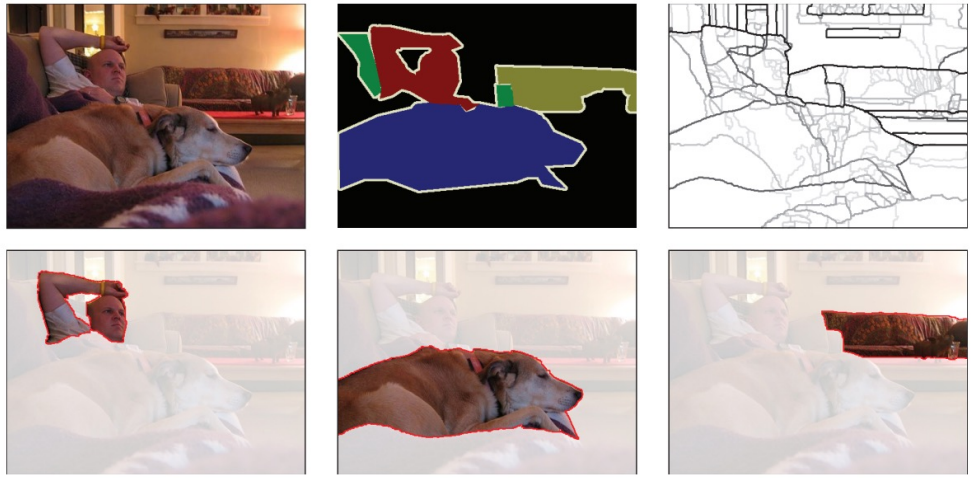


Figure 1. **Top:** original image, instance-level groundtruth from PASCAL and our multiscale hierarchical segmentation. **Bottom:** our best object candidates among 400.

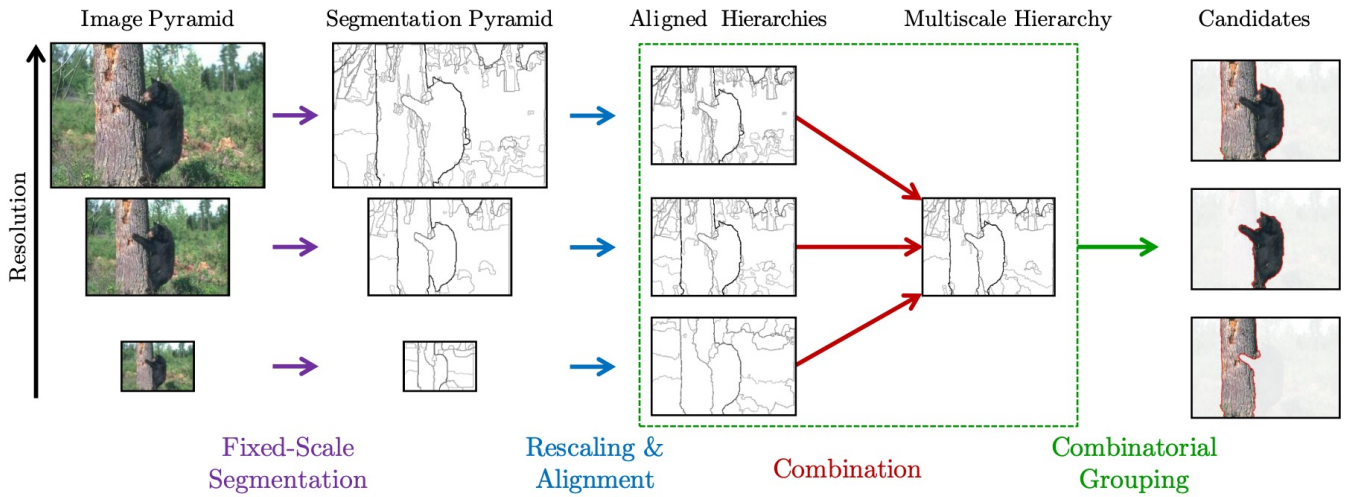


Figure 2. **Multiscale Combinatorial Grouping**. Starting from a multiresolution image pyramid, we perform hierarchical segmentation at each scale independently. We align these multiple hierarchies and combine them into a single multiscale segmentation hierarchy. Our grouping component then produces a ranked list of object candidates by efficiently exploring the combinatorial space of these regions.

2. 窗口评分的 proposal 方法（Window scoring proposal methods）

Window scoring proposal methods 通过对每个候选的窗口根据它们包含目标的概率来打分来产生 proposals 。与 grouping approaches 比，这些方法值返回边界框（bounding boxes），因此速度更快。但是，除非它们的窗口采样密度很高，否则这些方法位置精度很低。

• **Objectness** [12], [24]：最为最早和最广泛的一种 proposal 方法。它通过选择一副图片中的显著性位置作为 proposal，接着通过颜色，边缘，位置，尺寸，和 superpixel straddling 等多个线索对这些 proposal 打分。
• **Rahtu** [25]：以 一个包含采样区域（单个，两个和三个超像素）和 多个随机采样的框的大的 proposal 池作为开始。采用类似于 Objectness 的打分策略，但是有些提高 （[40]添加了额外的 low-level features 和 强调了恰当调优的非最大抑制（properly tuned nonmaximum suppression）的重要性）。
• **Bing**† [18]：通过边缘训练一个简单的线性分类器，并且以一个滑动窗口的方式运行。使用充足的近似，获得一个非常快的类未知的检测子 （CUP中每帧 1ms）。CrackingBing [41]表明一个有很小影响和类似性能的分类器可以通过不用查看图片的方式来获得 （**分类性能不是来自于学习而是几何学**）。
• **EdgeBoxes**† EC [20] ：基于目标边界估计（通过 structured decision forests [36], [42]获得）形成一个粗糙的滑动窗口模式作为开始，使用一个后续的 refinement 步骤来提高位置精度。不学习参数。作者提出通过调节滑动窗口模式的密度和和非最大抑制的阈值来调优方法用于不同的重叠阈值。
• **Feng** [43] ：通过搜索显著性图片内容来找到 proposal ，提出了一种新的显著性度量，包括一个潜在的目标能被图片的剩余部分组成。它采用滑动窗口模式，并通过显著性线索对每个位置打分。
• **Zhang** [44] ：提出在简单的梯度特征上训练一个级联的排序 SVMs。第一阶段对不同的尺度和长宽比（aspect ratio）训练不同的分类器；第二阶段对所有获得的proposals 排序。所有的 SVMs 使用结构性的输出，对含有更多目标重叠的窗口打分更高。因为级联在同样的类别上训练和测试，因此不太清楚它的泛化能力。
• **RandomizedSeeds** [45] ：使用多个随机的 SEED 超像素映射图 对每个候选窗口打分。打分策略类似于 Objectness 的 superpixel straddling （没有额外添加的信息）。作者展示使用多个超像素映射（superpixel maps ）可以明显地提高召回率。

B. Alexe, T. Deselaers, and V. Ferrari, “What is an object?” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2010, pp. 73–80.

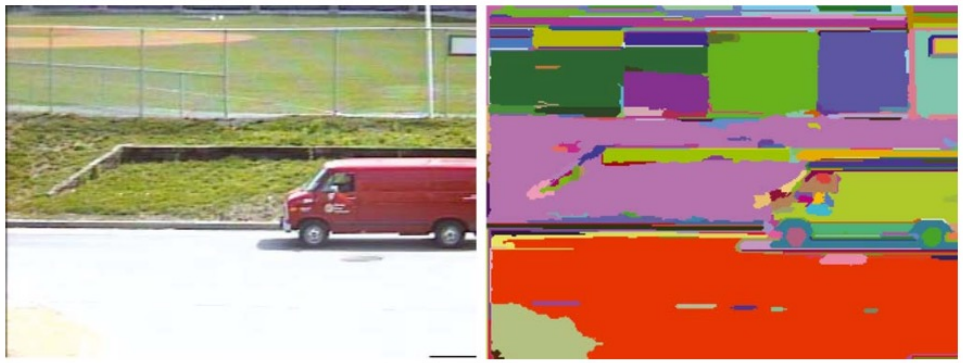


Figure 2. A street scene (320 × 240 color image), and the segmentation results produced by our algorithm ($\sigma = 0.8$, $k = 300$).

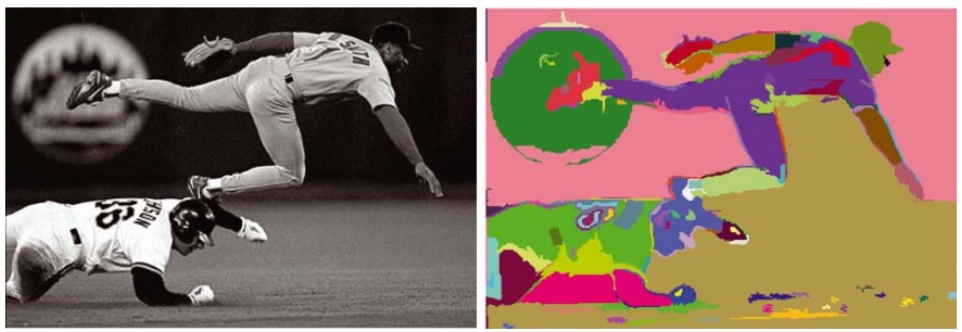


Figure 3. A baseball scene (432 × 294 grey image), and the segmentation results produced by our algorithm ($\sigma = 0.8$, $k = 300$).

Figure 4. An indoor scene (image 320 × 240, color), and the segmentation results produced by our algorithm ($\sigma = 0.8$, $k = 300$).

Objectness [12], [24]：最为最早和最广泛的一种 proposal 方法。它通过选择一副图片中的显著性位置作为 proposal，接着通过颜色，边缘，位置，尺寸，和 superpixel straddling 等多个线索对这些 proposal 打分。

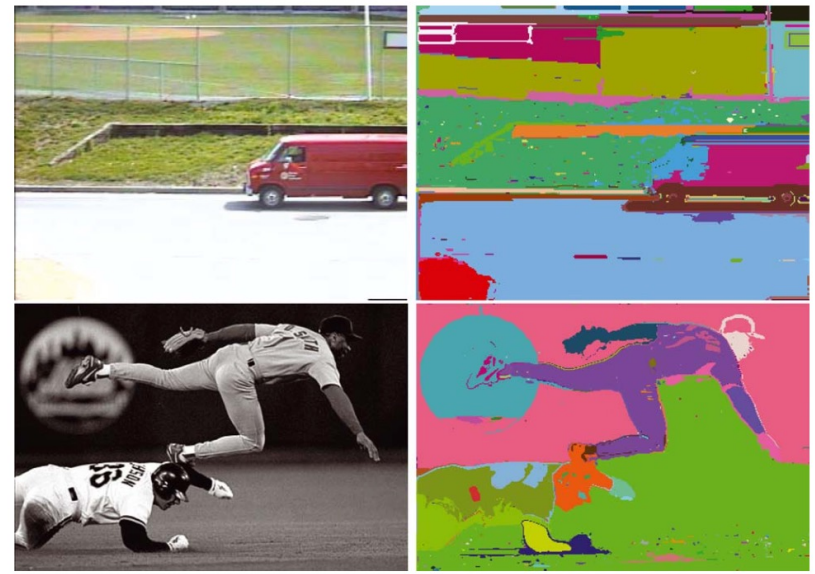


Figure 7. Segmentation of the street and baseball player scenes from the previous section, using the nearest neighbor graph rather than the grid graph ($\sigma = 0.8$, $k = 300$).

Figure 8. Segmentation using the nearest neighbor graph can capture spatially non-local regions ($\sigma = 0.8$, $k = 300$).

E. Rahtu, J. Kannala, and M. Blaschko, “Learning a category independent object detection cascade,” in Proc. IEEE Int. Conf. Comput. Vis., 2011, pp. 1052–1059.

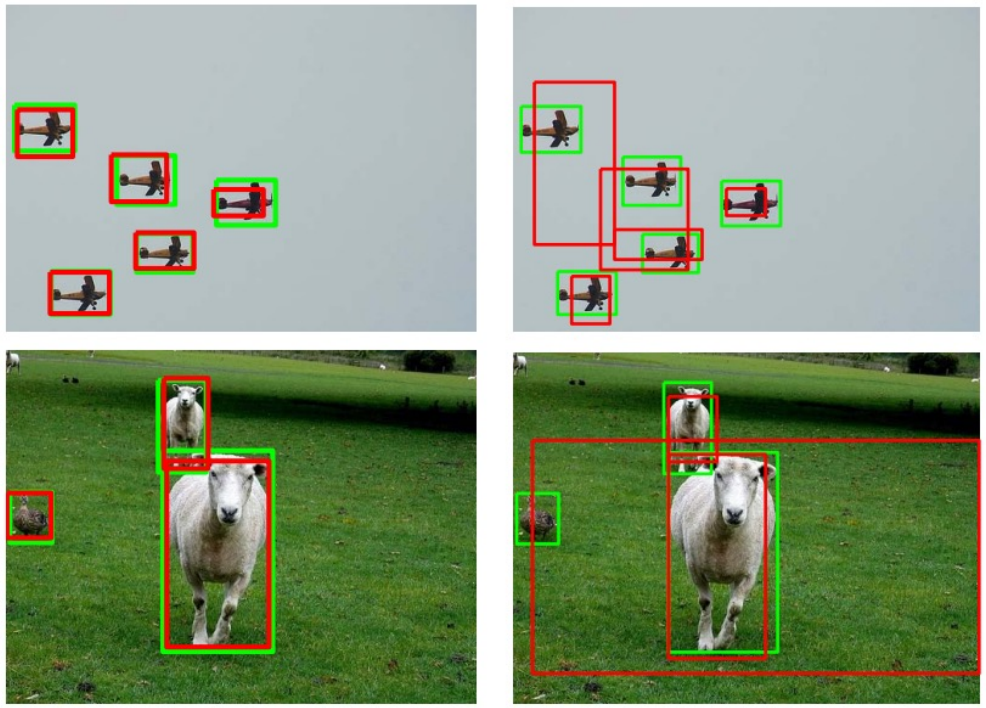


Figure 1. Example detections when returning 100 boxes with the proposed method (left) and the method by Alexe et al. [1] (right). The best detection for each ground-truth box (green) is shown.

Figure 3. Left: An image and an annotated bounding box. Middle: Superpixel segmentation. Right: A smoothed version of a binary image that shows the bounding boxes of superpixels.

Figure 4. Left: Original image. Right: Edge-weighted gradient magnitude maps for four main orientations.

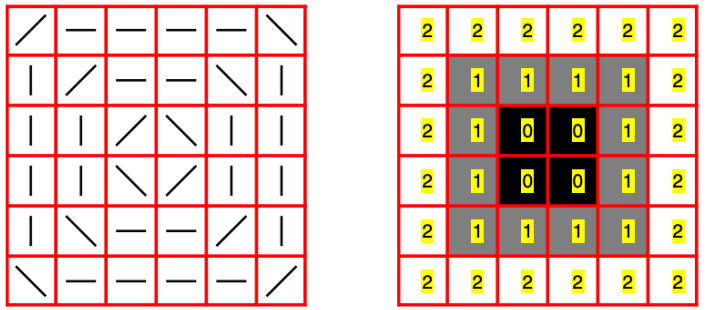


Figure 5. Window partition into 36 subregions. Left: Normal vector orientations for gradients considered in each subregion. Right: The weights for gradient magnitudes (γ_l).

Rahtu [25]：以一个包含采样区域（单个，两个和三个超像素）和 多个随机采样的框的大的 proposal 池作为开始。采用类似于 Objectness 的打分策略，但是有些提高 （[40]添加了额外的 low-level features 和 强调了恰当调优的非最大抑制（properly tuned nonmaximum suppression）的重要性）。

C. Zitnick and P. Dollar, “Edge boxes: Locating object proposals from edges,” in Proc. 13th Eur. Conf. Comput. Vis., 2014, pp. 391405.

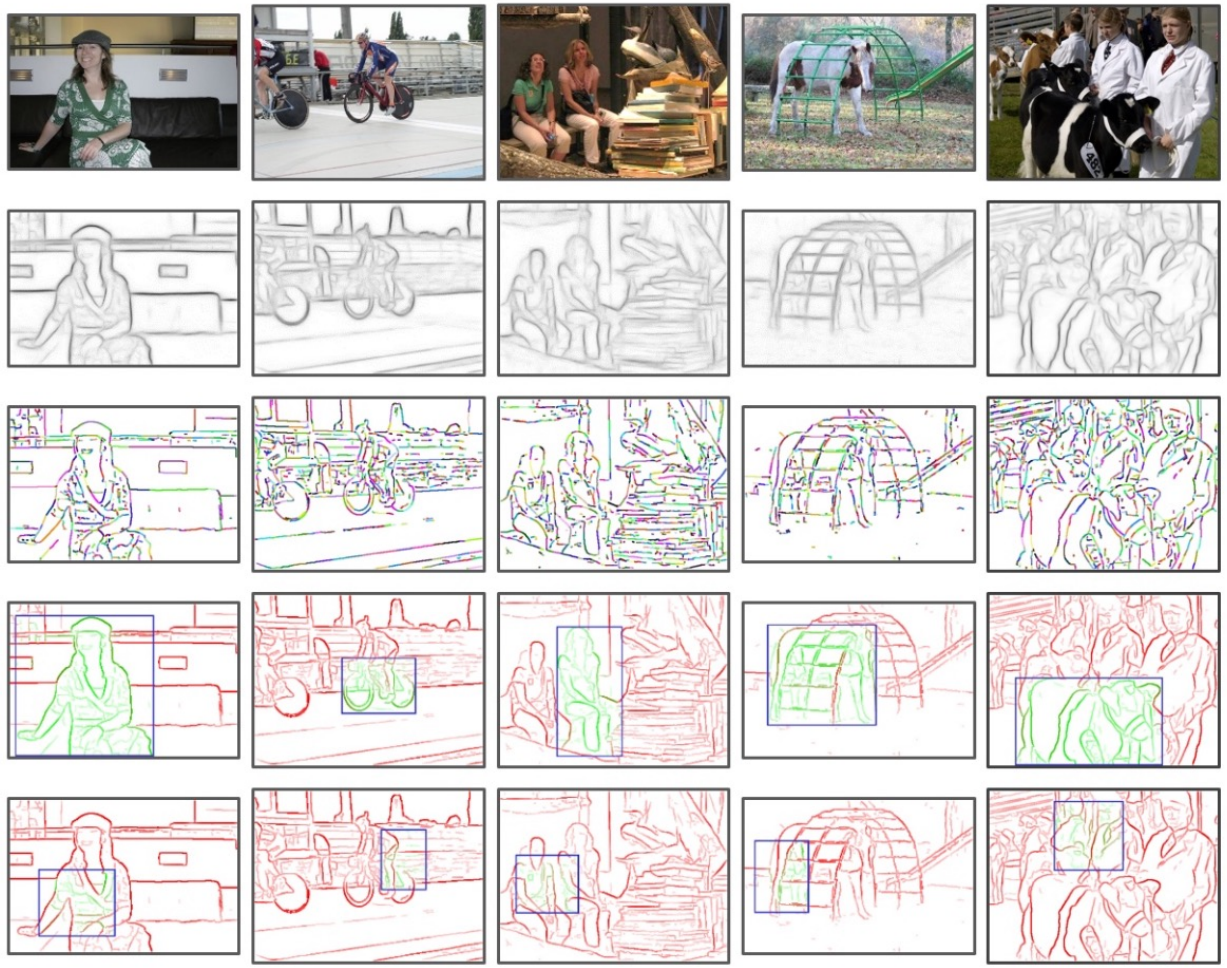


Fig. 1. Illustrative examples showing from top to bottom (first row) original image, (second row) Structured Edges [16], (third row) edge groups, (fourth row) example correct bounding box and edge labeling, and (fifth row) example incorrect boxes and edge labeling. Green edges are predicted to be part of the object in the box ($w_b(s_i) = 1$), while red edges are not ($w_b(s_i) = 0$). Scoring a candidate box based solely on the number of contours it *wholly encloses* creates a surprisingly effective object proposal measure. The edges in rows 3-5 are thresholded and widened to increase visibility.

EdgeBoxes† EC [20] ：基于目标边界估计（通过 structured decision forests [36], [42]获得）形成一个粗糙的滑动窗口模式作为开始，使用一个后续的 refinement 步骤来提高位置精度。不学习参数。作者提出通过调节滑动窗口模式的密度和和非最大抑制的阈值来调优方法用于不同的重叠阈值。

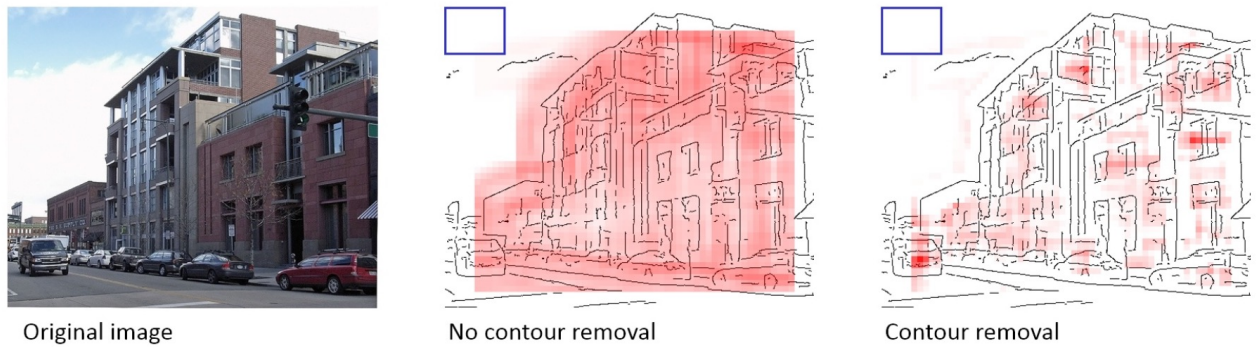


Fig. 3. Illustration of the computed score using (middle) and removing (right) contours that overlap the bounding box boundary. Notice the lack of clear peaks when the contours are not removed. The magnitudes of the scores are normalized for viewing. The box dimensions used for generating the heatmaps are shown by the blue rectangles.

J. Feng, Y. Wei, L. Tao, C. Zhang, and J. Sun, “Salient object detection by composition,” in Proc. IEEE Int. Conf. Comput. Vis., 2011, pp. 1028–1035.

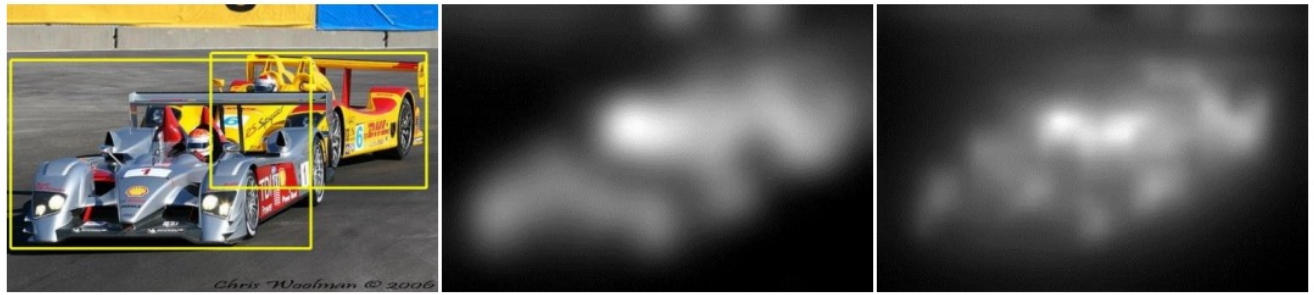


Figure 1. Left: image and salient objects found by our approach. Middle and right: two saliency maps of methods [8] and [6] generated using source code from [6].

Feng [43] ：通过搜索显著性图片内容来找到 proposal， 提出了一种新的显著性度量，包括一个潜在的目标能被图片的剩余部分组成。它采用滑动窗口模式，并通过显著性线索对每个位置打分。

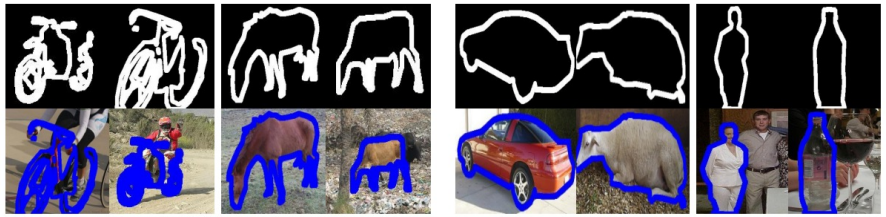
3 其他 proposal 方法（Alternative proposal methods）

• **ShapeSharing** [47] ：是一个无参的数据驱动的方法，通过匹配边转换目标形状从范例（exemplars）到测试图片。生成的区域使用图切割合并和提纯。

• **Multibox** [9], [48] ：训练一个神经网络来直接回归一定数量的 proposals （不需要在图片上滑动网络）。每个 proposals 都有它自己的位置误差 。该方法在 ImageNet 表现出最好的结果。

J. Kim and K. Grauman, "Shape sharing for object segmentation," in Proc. 12th Eur. Conf. Comput. Vis., 2012, pp. 444–458.

ShapeSharing [47] ：是一个无参的数据驱动的方法，通过匹配边转换目标形状从范例（exemplars）到测试图片。生成的区域使用图切割合并和提纯。



(a) Semantically *close* (b) Semantically *disparate*

Fig. 1. Intuition for shape sharing. While one may expect shape sharing between objects of semantically close categories (a), we observe that similar shapes exist even among semantically disparate objects (b). This suggests transferring "object-level" shapes between categories, to enable *category-independent* shape priors.

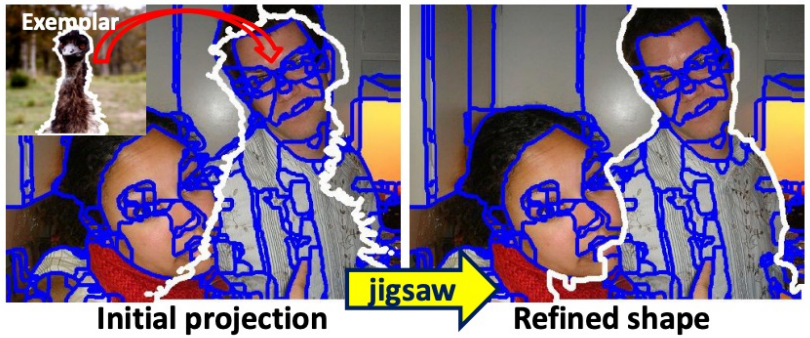


Fig. 3. Jigsaw puzzling the superpixels underlying the exemplar's projection

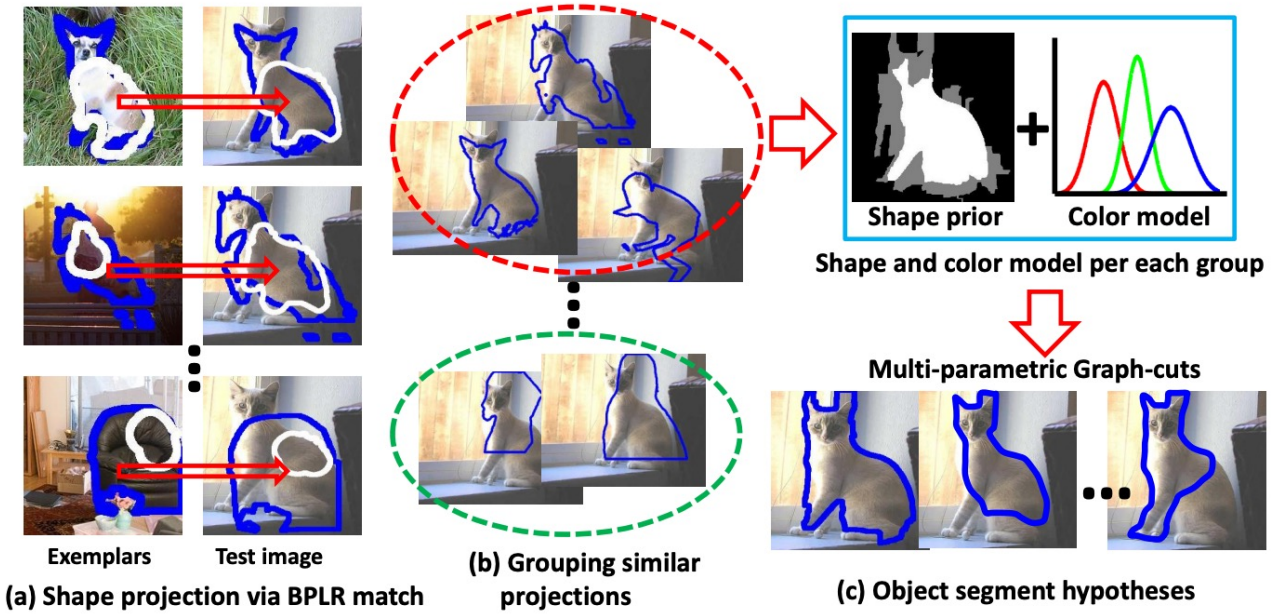


Fig. 2. Overview of our method. (a) Exemplars (first column) that share shape with the test image (second column) are projected in, no matter their category. We identify those shared shapes (marked in blue) via local BPLR matches (marked in white). (b) Multiple exemplars that highly overlap are aggregated, to form a shape prior and color model per each aggregated group. (c) The priors from each group are used to compute a series of graph-cut segmentation hypotheses.

TABLE 1
Comparison of Different Detection Proposal Methods

Method	Approach	Outputs Segments	Outputs Score	Control #proposals	Time (sec.)	Repeatability	Recall Results	Detection Results
`Bing` [18]	Window scoring		✓	✓	0.2	★★★	★	·
`CPMC` [19]	Grouping	✓	✓	✓	250	-	★★	★
`EdgeBoxes` [20]	Window scoring		✓	✓	0.3	★★	★★★	★★★
`Endres` [21]	Grouping	✓	✓	✓	100	-	★★★	★★
`Geodesic` [22]	Grouping	✓		✓	1	★	★★★	★★
`MCG` [23]	Grouping	✓	✓	✓	30	★	★★★	★★★
`Objectness` [24]	Window scoring		✓	✓	3	·	★	·
`Rahtu` [25]	Window scoring		✓	✓	3	·	·	★
`RandomizedPrim's` [26]	Grouping	✓		✓	1	★	★	★★
`Rantalankila` [27]	Grouping	✓		✓	10	★★	·	★★
`Rigor` [28]	Grouping	✓		✓	10	★	★★	★★
`SelectiveSearch` [29]	Grouping	✓	✓	✓	10	★★	★★★	★★★
`Gaussian`				✓	0	·	·	★
`SlidingWindow`				✓	0	★★★	·	·
`Superpixels`		✓			1	★	·	·
`Uniform`				✓	0	·	·	·

Grey check-marks indicate that the number of proposals is controlled by indirectly adjusting parameters. Repeatability, quality, and detection rankings are provided as rough summary of the experimental results: "-" indicates no data, "·", "★", "★★", "★★★" indicate progressively better results. These guidelines were obtained based on experiments presented in sections Section 3, Section 4, and Section 5, respectively.