A Hybrid Approach for Object Proposal Generation

Muhammd Aamir, Yi-Fei Pu, Waheed Ahmed Abro, Hamad Naeem, and Ziaur Rahman

1 Introduction

Recent years have witnessed a rapid evolution in computer vision and machine learning, with much effort being invested to enable machines to "see." Major road blocks have been solved, such as detecting edges in an image, segmenting images in more accurate ways, and learning different image features. The first step toward enabling machines to "see" is to enable a computer to recognize objects – which is the foundation of the visual world.

Object class detection is one of the key problems present in computer vision. While a human can easily recognize and detect objects, machines and computers still struggle due to diverse viewpoint variations like size, angle, perspective, occlusion, and illumination. In recent years, several approaches to object detection have been proposed to overcome these variations. A traditional approach for object detection is the sliding window approach, where the classifier is applied at every object location and scale. However, Girshick et al. [3] revolutionized this approach when he demonstrated a two-phase process method. In Girshick's process, a set of object proposals is first generated using a FAST algorithm, and then post-classification deep convolutional network classifier is applied on each of the proposals. This approach provides dramatic improvements in object detection accuracy as compared to the sliding window approach. Since Girshick's revolutionary demonstration, most current state-of-the-art object detectors have followed Girshick's lead and use object proposals as a first preprocessing step.

M. Aamir (✉) · Y.-F. Pu · H. Naeem · Z. Rahman
College of Computer Science Sichuan University, Chengdu, Sichuan, China

W. A. Abro
School of Computer Science and Engineering, Southeast University, Nanjing, China

M. Jiang et al. (eds.), *The Proceedings of the International Conference on Sensing and Imaging*, Lecture Notes in Electrical Engineering 506,
https://doi.org/10.1007/978-3-319-91659-0_18

Object detection performance depends upon both the object proposal algorithms and the post-classification detection networks. Merely improving post-classification, while beneficial, is not sufficient on its own. It is necessary for any post-classification improvements to be combined with a reduction in the number of image locations in order to be significant. Reducing image proposal not only speeds up object detection but also reduces the false positives in the post-classification stage. The goal is to reduce the number of proposals at the generation time in order to be used in real-time applications more efficiently and to automatically generate a small number of diverse regions that may contain objects in an image. Each object of an image must be well represented in at least one region.

In this paper, we propose a new hybrid object proposal method which significantly reduces the number of proposals generated and the number of false positives in the post-classification phase. We first get initial proposals from hierarchical segmentations [1] and then rank the proposals as per score criteria. Scoring regions is done using contours enclosed in the region, and then top of object proposals passes for post-classification.

2 Related Works

In this section, we concisely review previous approaches to object detection, most of which use object classifiers and object proposal algorithms. These methods are broadly divided into two categories: groping methods and window scoring methods. Grouping methods generate multiple segments of an image which are likely to contain objects. The most common approach to grouping methods is to do hierarchical image segmentation and merge segments according to the similarities between those segments. Most grouping algorithm performance relies on initial segmentation algorithms. Felzenszwalb [4] algorithm is well suited for this purpose, as his algorithm is both efficient and timely. Algorithms generate set of small initial regions at a rapid speed, which, in turn, define segmentation as graph problems where each vertex is an element to be segmented, and edges are between two neighboring regions. Algorithms then make region comparisons, each segment corresponding to a connected component of the graph.

Carreira and Sminchisescu [5], CMPC, and Endres and Hoiem [6] methods solve multiple graph cuts with different seeds and parameters to generate class-independent proposals. Both of these methods generate binary foreground/background segments, with each obtained foreground segment as an object hypothesis, and both of these methods learn to predict the segments that cover complete objects and rank proposals accordingly. However, both algorithms are slow due to their reliance on the gPb edge detector but generate high-quality segmentation masks. Selective search [1] method is the most widely used method in object recognition and object detection and is based on multiple hierarchical segmentation using superpixels. For covering a diverse set of regions, different kinds of grouping strategies and color spaces are used which produces high recall at fast speeds – a few seconds per image. However, there is no scoring mechanism on the proposals; therefore, proposals cannot be ranked.

Table 1 The performance comparisons of both approaches are given in the chart below

Methods	Approach	Output segments	Output score	Time (s)
Selective search [1]	Grouping	Yes	No	10
CPMC [5]	Grouping	Yes	Yes	250
Endres and Hoiem [6]	Grouping	Yes	Yes	100
Rantalankila [7]	Grouping	Yes	No	10
Objectness [8]	Window scoring	No	Yes	3
Rahtu [9]	Window scoring	No	Yes	3
EdgeBox [2]	Window scoring	No	YES	0.3
Bing [10]	Window scoring	No	YES	0.2

On the other hand, window scoring methods is very different, with each window score being calculated according to how likely it is to contain an object. This approach generates a bounding box much faster than the grouping methods. However, this approach has low localization accuracy. Objectness [8] is a window-based approach in which each candidate window score is calculated on different image cues. Objectness stands as one of one of the earliest object proposal methods and is capable of measuring the likelihood that objects are present in the image. This method uses saliency, color contrast, edge density, and superpixel straddling cues to obtain characteristics of images and adopts Bayesian's framework to combine several cues. This has shown that the new combined cues outperform the state-of-the-art saliency measure. The last advantage of objectness is its slow emergence of drawback, which appears at a snail's speed. This method has low localization accuracy, but the first few proposals it obtains are of high quality.

EdgeBox [2] is another window-based approach and is among the fastest object proposal generation methods. EdgeBox generates object proposals directly from the edges of an image. Initial edge maps are computed from edge detectors [11] and then are combined into eight connecting edges to form an edge group. This method uses sliding window search over a scale to generate a candidate box and then scores each box, selecting the top few thousand proposals. Rahtu et al. [9] begins with a large number of randomly sampled boxes from an objectness and multiplies them with proposal regions generated from single, pair, and triplet superpixel segmentations. And their score function is similar to that of objectness, where they have made some improvements by adding low-level features (Table 1).

Girshick et al. [3] introduced their R-CNN method which defines object detection in a two-step process. This method generates a set of category-independent proposals using bottom-up grouping (i.e., selective search). Girshick et al. then used a deep convolutional neural network on those generated proposals. This method dramatically improves the performance proposal generation, proposal classification, and overall object detection by replacing the traditional sliding window approach with object proposals, thus achieving a state-of-the-art object detection performance. Fast R-CNN [8] is an improvement of Girshick et al.'s previous work and allows for faster object detection.

This paper presents a hybrid approach which combines both grouping and window scoring methods to increase the detection performance. This hybrid method results in excellent object detection task completion at relatively fast speeds compared to selective search methods and greatly reduces the false-positive rate.

3 Proposed Work

In this paper, we have proposed a new hybrid object proposal approach which combines hierarchical segmentations [1] and window scoring method [2]. First, we generate object proposals through the agglomerative clustering grouping method. We then score the boxes according to the sums of the magnitude of the all the edges in each edge group minus the edge groups of the contours that straddle the bounding box. Finally, we rank the object proposals according to score of the boxes. The top-ranked proposals can then be chosen for the classification task. However, there is still a great deal of importance in reducing object proposal generation time, as it also reduces the false-positive rate.

We observed that R-CNN achieves object detection at a faster rate due to reducing object location – from all locations to proposed location – while the object proposal generated by selective search [1] was still very high (around 8–10 thousand). Furthermore, we have reduced object proposals by ranking object proposal according to box score and only have select top few thousand proposals for object detection.

3.1 Algorithm Overview

The major steps of our algorithm are as follows:

1. Segmentation: Our proposal begins by generating a set of initial regions on which we apply hierarchical clustering.
2. Hierarchical Clustering: We group initial segments obtained from the above step according by the similarity measure between neighboring regions.
3. Edge detection and edge groups: We generate image edge maps with the structured edge detector. And, from edge map, we form edge groups by grouping neighboring edges according to orientation similarity.
4. Score regions: Regions obtained from clustering are forwarded for scoring. We score regions according to the strength of the edges in the edge groups within the region and then subtract the strength of edges in the edge groups that straddle the region.
5. Ranking: We rank the proposal according to score of the region.

Segmentation

As most of the grouping methods generate object proposals using segmentation, we also use segmentation to obtain a small set of starting regions for hierarchical clustering. We use Felzenszwalb and Huttenlocher's graph-based algorithm, which is an efficient method for obtaining regions. This method is well suited for our purpose because of its speed and accuracy. It converts images into a graph – pixels are the vertices and neighboring pixels are connected with the edges. We then manipulate the graph to segment the image.

Hierarchical Clustering

Regions obtained from step 1 serve as starting points for hierarchical clustering. Agglomerative (bottom-up) clustering method is then used, where initially each region is a cluster. We repeatedly combine two similar neighboring regions – after each combination new similarities are calculated. This process continues until the whole image becomes one cluster/region. We then use color, texture, size, and gap similarity measures. Hierarchical clustering is applied on different color spaces to cover a more diverse set of regions. Regions from each hierarchy are then combined, while duplicate regions are removed at the end. Clusters obtained from hierarchical clustering are the object proposals; we repeat the clustering algorithm in different color spaces.

Edge Detection and Edge Group

For edge detection, we use structured edge detection. Structure forest extract image patches from the image, convert each image patch into vectors, extract the image features for each patch, and then predict scores of the patches at the edge. The edges obtained from detector are then combined into eight connected neighboring edges with similar orientation until the orientation differences above pi/2 form the edge groups. This method shows good accuracy and speed as compared to traditional edge detectors.

Score Regions

Given set object proposals obtained from hierarchical clustering, we calculate the score of each object proposal. This is accomplished by summing the magnitude of every wholly enclosed edge in the group in a given region and subtracting the magnitude of every edge in the group which straddles the object region. The value of $w_b(s_i)$ is calculated for each edge group to check if the group is wholly enclosed in the region. When an edge group is not wholly closed in the box, then $w_b(s_i) = 0$. If an edge group is wholly enclosed in the box $w_b(s_i)$ is calculated as below:

$$w_b\left(\mathrm{s}_i\right) = 1 - \max_t \prod\nolimits_j^{|\mathrm{T}|-1} a\left(t_j - t_{j+1}\right) \tag{1}$$

where “a” is the affinity and “t” is the order path, so the above equation finds the order path with the max affinity between the groups. We then compute the score using the formula:

$$h(b) = \frac{\sum_i w_b\left(s_i\right) m_i}{2(b_w + b_h)^k} \tag{2}$$

where b_w and b_n are the box width and height and k is the bias value for larger boxes.

Ranking

We rank objects proposed according to score obtained from Eq. 2, where a few thousands of object proposals passed for classification task (Figs. 1 and 2).

4 Evaluations and Results

Most of our experiments were performed on a PASCAL VOC 2007 dataset [12], which contains 9963 images, with a training set containing 2501 images, validation set containing 2510 images, and test set containing 4952 images. The dataset has 20 object classes in four broad categories – person, animal, vehicle, and indoor.

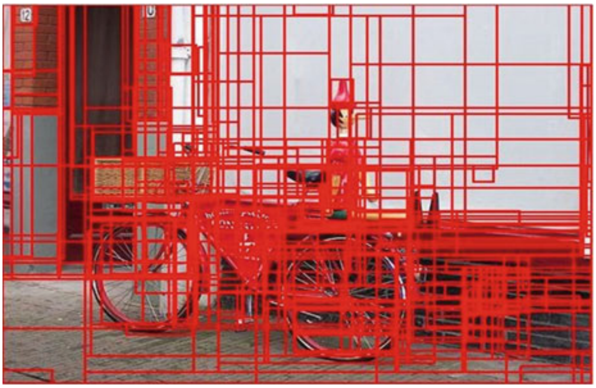

Fig. 1 Proposal evaluation on VOC

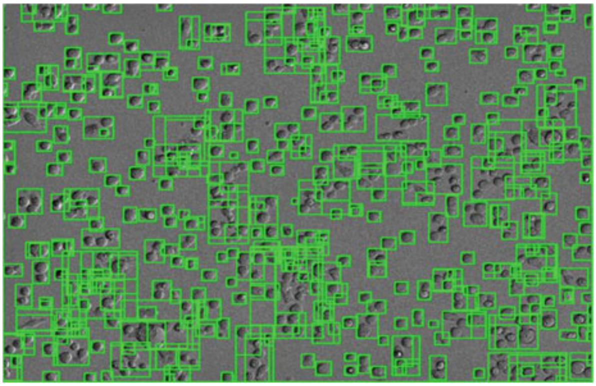

Fig. 2 Proposal evaluation on migrating cancer dataset

Training images are labeled with ground truth from 20 object classes. Every image has an annotation that contains the bounding box information and difficulty level of the object.

PASCAL VOC provides standardized images, which contain a large number of objects and a cornucopia of categories, scales, illuminations, viewpoints, and positions – making this database ideal for object reorganization. PASCAL 20 visual object classes are airplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, dining table, dog, horse, motorbike, person, potted plant, sheep, sofa, train and TV monitor. We have performed all our experiments on a CPU with 4GB RAM. For evaluating the quality of our object proposals, we use two measures: ABO (average best overlap) and MABO (mean average best overlap).

4.1 Average Best Overlap (ABO)

Average best overlap, for any class, is achieved by calculating best overlap on ground truth of class and proposed object region of said class and then taking its average. Overlap is the intersection of proposed region with ground truth over area of their union.

$$\text{IoU}\ \ (\text{box}, \text{gtruth}) = \frac{\text{area}\,(\text{box}) \cap \text{area}\ \ (\text{gtruth})}{\text{area}\,(\text{box}) \cup \text{area}\ \ (\text{gtruth})}$$

4.2 Mean Average Best Overlap (MABO)

Mean average best overlap, is the mean ABO over all classes. We have evaluated our proposal on PASCAL VOC 2007 test set and compare with selective search and edge box proposal generation methods (Tables 2 and 3).

Table 2 Mean average best overlap on VOC dataset

Methods	Test images	Proposals	MABO (mean average best overlap)
Edge box	4952	1500	0.799
Selective search	4952	1500	0.820
Our proposal	4952	1500	0.833

Table 3 Average best overlap for 20 classes of VOC on top 1500 proposals

VOC classes	Edge box ABO	Selective search ABO	Our proposal ABO
Plane	0.771	0.796	0.807
Bicycle	0.824	0.844	0.861
Bird	0.796	0.812	0.812
Boat	0.779	0.768	0.784
Bottle	0.692	0.660	0.673
Bus	0.841	0.864	0.868
Car	0.788	0.783	0.808
Cat	0.827	0.906	0.909
Chair	0.783	0.798	0.808
Cow	0.827	0.829	0.854
Table	0.817	0.891	0.894
Dog	0.837	0.895	0.900
Horse	0.815	0.828	0.841
Bike	0.815	0.829	0.846
Person	0.755	0.754	0.766
Potted plant	0.746	0.740	0.758
Sheep	0.814	0.797	0.828
Sofa	0.828	0.904	0.907
Train	0.801	0.856	0.863
TV monitor	0.821	0.842	0.868

5 Conclusions and Future Work

In summary, our efficient, new hybrid method for generating object proposals uses selective search proposal and scores them according to edges present in the proposed regions. This method results in adequate detection rates for object detection task – compared to object detection solely utilizing selective search – and significantly decreases the false-positive rate. Throughout this paper, we demonstrate that our purposed hybrid method matches the accuracy of selective search, with only 25% the number of proposal after ranking said proposals. Our method results in high-quality class-independent object locations, with mean average best overlap of 0.833 at 1500 locations.

In the future, the score function can be further optimized by penalizing the portion of edge groups that overlap the region boundary, instead of subtracting strength of edges present in edge group. The edge box generates redundant object proposals in each scale; therefore, by reducing redundant object proposals, edge box performance can also be further improved. Furthermore, we can use a strong post-classification, deep convolutional features and strong appearance models for object detection with reduced object proposals.

Acknowledgments This work was supported by the National Natural Science Foundation of China under grants 61571312, Academic and Technical Leaders Support Foundation of Sichuan province under grants (2016)183-5, and National Key Research and Development Program Foundation of China under grants 2017YFB0802300. The authors would like to thank Ms. Siobhan Kathryn He for the constructive criticism of the manuscript.

References

1. Uijlings JRR, van de Sande KEA, Gevers T, Smeulders AWM (2013) Selective search for object recognition. Inter J Comp Vision 104(2):154–171
2. Zitnick CL, Dollár P (2014) Edge boxes: locating object proposals from edges. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds) Computer vision – ECCV 2014. Lecture notes in computer science, vol 8693. Springer, Cham
3. Girshick RB, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE conference on computer vision and pattern recognition, pp 580–587
4. Felzenszwalb PF, Huttenlocher DP (2004) Efficient graph-based image segmentation. Inter J Comp Vision 59(2):167–181
5. Carreira J, Sminchisescu C (2012) Cpmc: automatic object segmentation using constrained parametric min-cuts. PAMI 34(7):1312
6. Endres I, Hoiem D (2014) Category-independent object proposals with diverse ranking. PAMI 36:222
7. Rantalankila P, Kannala J, Rahtu E (2014) Generating object segmentation proposals using global and local search. In: Computer vision and pattern recognition (CVPR), 2014 IEEE conference on. IEEE, pp 2417–2424
8. Alexe B, Deselaers T, Ferrari V (2012) Measuring the objectness of image windows. PAMI 34(11):2189
9. Rahtu E, Kannala J, Blaschko M (2011) Learning a category independent object detection cascade. In: Computer vision (ICCV), 2011 IEEE international conference on. IEEE, pp 1052–1059
10. Cheng M-M, Zhang Z, Lin W-Y, Torr P (2014) BING: Binarized normed gradients for objectness estimation at 300fps. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3286–3293
11. Dollar P, Zitnick CL (2014) Fast edge detection using structured forests. CoRR abs/1406.5549
12. Everingham M, Ali Eslami SM, Van Gool L, Williams CKI, Winn J, Zisserman A (2015) The pascal visual object classes challenge: a retrospective. Inter J Comp Vision 111(1):98–136