Attend Refine Repeat: Active Box Proposal Generation via In-Out Localization

Spyros Gidaris
spyros.gidaris@enpc.fr
Nikos Komodakis
nikos.komodakis@enpc.fr

Université Paris-Est, École des Ponts ParisTech
Paris, France

Abstract

The problem of computing category agnostic bounding box proposals is utilized as a core component in many computer vision tasks and thus has lately attracted a lot of attention. In this work we propose a new approach to tackle this problem that is based on an active strategy for generating box proposals that starts from a set of seed boxes, which are uniformly distributed on the image, and then progressively moves its attention on the promising image areas where it is more likely to discover well localized bounding box proposals. We call our approach *AttractioNet* and a core component of it is a CNN-based category agnostic object location refinement module that is capable of yielding accurate and robust bounding box predictions regardless of the object category. We extensively evaluate our *AttractioNet* approach on the COCO 2014 validation set as well as on the PASCAL VOC2007 test set, reporting for both of them state-of-the-art results that surpass the previous work in the field by a significant margin. Finally, we provide strong empirical evidence that our approach is capable to generalize to unseen categories. Project page:: https://github.com/gidariss/AttractioNet.

1 Introduction

Category agnostic object proposal generation is a computer vision task that has received an immense amount of attention over the last years. Its definition is that for a given image a small set of instance segmentations or bounding boxes must be generated that will cover with high recall all the objects that appear in the image regardless of their category. In object detection, applying the recognition models to such a reduced set of category independent location hypothesis [16] instead of an exhaustive scan of the entire image [11, 34], has the advantages of drastically reducing the amount of recognition model evaluations and thus allowing the use of more sophisticated machinery for that purpose. As a result, proposal based detection systems manage to achieve state-of-the-art results and have become the dominant paradigm in the object detection literature [3, 13, 14, 15, 16, 20, 33, 36, 40]. Object proposals have also been used in various other tasks, such as weakly-supervised object detection [7], exemplar 2D-3D detection [29], visual semantic role labelling [18], caption generation [23] or visual question answering [35].

In this work we focus on the problem of generating bounding box object proposals rather than instance segmentations. Several approaches have been proposed in the literature for

Pages 90.1-90.13
DOI: https://dx.doi.org/10.5244/C.30.90

this task [1, 2, 5, 6, 8, 19, 24, 25, 28, 38, 43]. Among them our work is most related to the CNN-based objectness scoring approaches [12, 26, 32] that recently have demonstrated state-of-the-art results [31, 32].

In the objectness scoring paradigm, a large set of image boxes is ranked according to how likely it is for each image box to tightly enclose an object — regardless of its category — and then this set is post-processed with a non-maximum-suppression step and truncated to yield the final set of object proposals. In this context, Kuo *et al*. [26] with their DeepBox system demonstrated that training a convolutional neural network to perform the task of objectness scoring can yield superior performance over previous methods that were based on low level cues and they provided empirical evidence that it can generalize to unseen categories. In order to avoid evaluating the computationally expensive CNN-based objectness scoring model on hundreds of thousands image boxes, which is necessary for achieving good localization of all the objects in the image, they use it only to re-rank the proposals generated from a faster but less accurate proposal generator thus being limited by its localization performance. Instead, more recent CNN-based approaches apply their models only to ten of thousands image boxes, uniformly distributed in the image, and jointly with objectness prediction they also infer the bounding box of the closest object to each input image box. Specifically, the Region Proposal Network in Faster-RCNN [33] performs bounding box regression for that purpose while the DeepMask method predicts the foreground mask of the object centred in the image box and then it infers the location of the object's bounding box by extracting the box that tightly encloses the foreground pixels. The latter has demonstrated state-of-the-art results and was recently extended with a top-down foreground mask refinement mechanism that exploits the convolutional feature maps at multiple depths of a neural network [31].

Our work is also based on the paradigm of having a CNN model that given an image box it jointly predicts its objectness and a new bounding box that is better aligned on the object that it contains. However, we opt to advance the previous state-of-the-art in box proposal generation in two ways: **(1)** improving the object's bounding box prediction step **(2)** actively generating the set of image boxes that will be processed by the CNN model.

Regarding the bounding box inference step we exploit the recent advances in object detection where Gidaris and Komodakis [14] showed how to improve the object-specific localization accuracy. Specifically, they replaced the bounding box regression step with a localization module, called *LocNet*, that given a search region it infers the bounding box of the object inside the search region by assigning membership probabilities to each row and each column of that region and they empirically proved that this localization task is easier to be learned from a convolutional neural network thus yielding more accurate box predictions during test time. Given the importance of having accurate bounding box locations in the proposal generation task, we believe that it would be of great interest to develop and study a *category agnostic* version of LocNet for this task.

Our second idea for improving the box proposal generation task stems from the following observation. Recent state-of-the-art box proposal methods evaluate only a relatively small set of image boxes (in the order of $10k$) uniformly distributed in the image and rely on the bounding box prediction step to fix the localization errors. However, depending on how far an object is from the closest evaluated image box, both the objectness scoring and the bounding box prediction for that object could be imperfect. For instance, Hosang *et al*. [21] showed that in the case of the detection task the correct recognition of an object from an image box is correlated with how well the box encloses the object. Given how similar are the tasks of category-specific object detection and category-agnostic proposal generation, it

is safe to assume that a similar behaviour will probably hold for the latter one as well. Hence, in our work we opt for an *active* object localization scheme, which we call *Attend Refine Repeat* algorithm, that starting from a set of seed boxes it progressively generates newer boxes that are expected with higher probability to be on the neighbourhood or to tightly enclose the objects of the image. Thanks to this localization scheme, our box proposal system is capable to both correct initially imperfect bounding box predictions and to give higher objectness score to candidate boxes that are more well localized on the objects of the image. Note that active localization schemes have also been previously applied in the object detection literature [4, 13, 14, 17, 30, 39, 42].

To summarize, our contributions with respect to the box proposal generation task are: **(1)** We developed a box proposal system that is based on an improved category-agnostic object location refinement module and on an active box proposal generation strategy that behaves as an attention mechanism that focus on the promising image areas in order to propose objects. We call the developed box proposal system *AttractioNet*: *(Att)end (R)efine Repeat: (Act)ive Box Proposal Generation via (I)n-(O)ut Localization (Net)work*. **(2)** We exhaustively evaluate our system both on PASCAL and on the more challenging COCO datasets and we demonstrate significant improvement with respect to the state-ofthe-art on box proposal generation. Furthermore, we provide strong evidence that our object location refinement module is capable of generalizing to unseen categories.

The remainder of the paper is structured as follows: We describe our box proposal methodology in section §2, we show experimental results in section §3 and we present our conclusions in section §4.

2 Our approach

2.1 Active bounding box proposal generation

Algorithm: ***Attend Refine Repeat***

Input : Image $\mathbf{I}$
Output: Bounding box proposals $\mathbf{P}$
$\mathbf{C} \leftarrow \emptyset$, $\mathbf{B}^0 \leftarrow$ seed boxes
for $t \leftarrow 1$ ***to*** T **do**
  `/* Attend & Refine procedure */`
  $\mathbf{O}^t \leftarrow \mathit{ObjectnessScoring}(\mathbf{B}^{t-1}|\mathbf{I})$
  $\mathbf{B}^t \leftarrow \mathit{ObjectLocationRefinement}(\mathbf{B}^{t-1}|\mathbf{I})$
  $\mathbf{C} \leftarrow \mathbf{C} \cup \{\mathbf{B}^t, \mathbf{O}^t\}$
end
$\mathbf{P} \leftarrow \mathit{NonMaxSuppression}(\mathbf{C})$

The active box proposal generation strategy that we employ in our work, which we call *Attend Refine Repeat* algorithm, starts from a set of seed boxes, which only depend on the image size, and it then sequentially produces newer boxes that will better cover the objects of the image while avoiding the "objectless" image areas (see Figure 1). At the core of this algorithm lies a CNN-based box proposal model that, given an image I and the coordinates of a box B, executes the following operations:

Category agnostic object location refinement: this operation returns the coordinates of a new box $\tilde{B}$ that would be more tightly aligned on the object near B. In case there are more than one objects in the neighbourhood of B then the new box $\tilde{B}$ should be

Figure 1: Illustration of image areas that are being attended by our active proposal generation algorithm as it progresses from the fist iteration (1st column) to the last one (5th column). Note that in order to create the provided attention maps we collapsed the attended boxes of each iteration in a 2D canvas. We observe that in the first iteration the box proposal generator attends the entire image since the seed boxes are created by uniformly distributing boxes across the image. However, as the algorithm progresses its attention is concentrated on the image areas that actually contain objects.

targeting the object closest to the input box B, where by closest we mean the object that its bounding box has the highest intersection over union (IoU) overlap with the input box B.

Category agnostic objectness scoring: this operation scores the box B based on how likely it is to tightly enclose an object, regardless of its category.

The pseudo-code of the *Attend Refine Repeat* algorithm is provided in Algorithm 1. Specifically, it starts by initializing the set of candidate boxes $\mathbf{C}$ to the empty set and then creates a set of seed boxes $\mathbf{B}^0$ by uniformly distributing boxes of various fixed sizes in the image (similar to Cracking Bing [41]). Then on each iteration t it estimates the objectness $\mathbf{O}^t$ of the boxes generated in the previous iteration, $\mathbf{B}^{t-1}$, and it refines their location (resulting in boxes $\mathbf{B}^t$) by attempting to predict the bounding boxes of the objects that are closest to them. The results $\{\mathbf{B}^t, \mathbf{O}^t\}$ of those operations are added to the candidates set $\mathbf{C}$ and the algorithm continues. In the end, non-maximum-suppression [11] is applied to the candidate box proposals $\mathbf{C}$ and the top K box proposals, set $\mathbf{P}$, are returned.

The advantages of having an algorithm that sequentially generates new box locations given the predictions of the previous stage are two-fold:

- **Attention mechanism:** First, it behaves as an attention mechanism that, on each iteration, focuses more and more on the promising locations (in terms of box coordinates) of the image (see Figure 1). As a result, boxes that tightly enclose the image objects are more likely to be generated and to be scored with high objectness confidence.
- **Robustness to initial boxes:** Furthermore, it allows to refine some initially imperfect box predictions or to localize objects that might be far (in terms of center location, scale and/or aspect ratio) from any seed box in the image. This is illustrated via a few characteristic examples in Figure 2. As shown in each of these examples, starting from a seed box, the iterative bounding box predictions gradually converge to the closest object without actually being affected from any nearby instances.

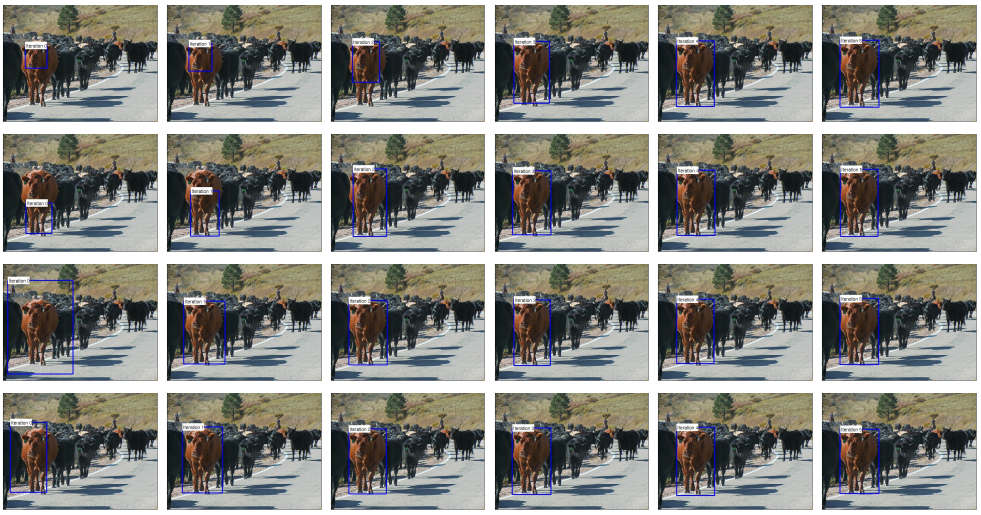


Figure 2: Illustration of the consecutive bounding box predictions made by our category agnostic location refinement module as the active proposal generation algorithm progresses from the 0th iteration (seed box; 1st column) to the last iteration (final box prediction; 6th column). Each row depicts a different example. Despite the fact that the seed box might be quite far from the object (in terms of center location, scale and/or aspect ratio) the refinement module has no problem in converging to the bounding box closest to the seed box object. This capability is not affected even in the case that the seed box contains also other instances of the same category as in rows 3 and 4.

2.2 CNN-based box proposal model

Here we describe in more detail the object localization and objectness scoring modules of our box proposal model as well as the CNN architecture that implements it.

2.2.1 Object location refinement module

In order for our active box proposals generation strategy to be effective, it is very important to have an accurate and robust category agnostic object location refinement module. Hence we follow the paradigm of the recently introduced LocNet model [14] that has demonstrated superior performance in the category specific object detection task over the typical bounding box regression paradigm [13, 15, 33, 34] by formulating the problem of bounding box prediction as a dense classification task. Here we use a properly adapted version of that model for the task at hand.

At a high level, given as input a bounding box B, the location refinement module first defines a search region $R = \gamma B$ (i.e., the region of B enlarged by a factor γ) over which it is going to next search for a new refined bounding box. To achieve this, it considers a discretization of the search region R into M columns as well as M rows, and yields two probability vectors, $p_x = \{p_{x,i}\}_{i=1}^{M}$ and $p_y = \{p_{y,i}\}_{i=1}^{M}$, for the M columns and the M rows respectively of R, where these probabilities represent the likelihood of those elements (rows or columns) to be inside the target box B^* (these are also called *in-out* probabilities in the original LocNet model). Each time the target box B^* is defined to be the bounding box of the object closest to the input box B. Finally, given those *in-out* probabilities, the object location $\tilde{B}$ inference is formulated as a simple maximum likelihood estimation problem that maximizes the likelihood of the *in-out* elements of $\tilde{B}$. A visual illustration of the above process through a few examples is provided in Fig. 3 (for further details about the LocNet model we refer the interested reader to [14]).

We note that in contrast to the original LocNet model that is optimized to yield a different set of probability vectors of each category in the training set, here our category-agnostic version is designed to yield a single set of probability vectors that should accurately localize any

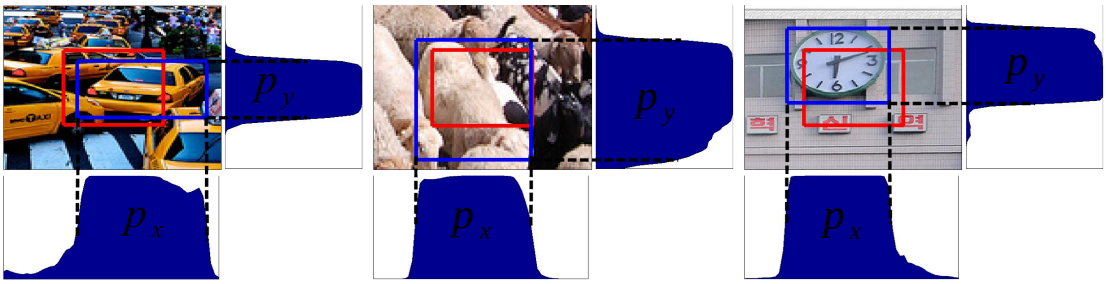


Figure 3: Illustration of the bounding box prediction process that is performed by our location refinement module. In each case the red rectangle is the input box B, the blue rectangle is the predicted box and the depicted image crops are the search regions where the refinement module "looks" in order to localize the target object. On the bottom and on the right side of the image crop we visualize the p_x and p_y probability vectors respectively that our location refinement module yields in order to localize the target object. Ideally, those probabilities should be equal to 1 for the elements (columns/rows) that overlap with the target box and 0 everywhere else.

object regardless of its category (see also section §2.2.3 that describes in detail the overall architecture of our proposed model). It should be also mentioned that this is a more challenging task to learn since, in this case, the model should be able to localize the target objects even if they are in crowded scenes with other objects of the same appearance and/or texture (see the two left-most examples of Figure 3) without exploiting any category supervision during training that would help it to better capture the appearance characteristics of each object category. On top of that, our model should be able to localize objects of unseen categories. In the right-most example of Figure 3, we provide an indicative result produced by our model that verifies this test case. In this particular example, we apply a category-agnostic refinement module trained on PASCAL to an object whose category ("clock") was not present in the training set and yet our trained model had no problem of confidently predicting the correct location of the object. In section 3.2 of the paper we also provide quantitative results about the generalization capabilities of the location refinement module.

2.2.2 Objectness scoring module

The functionality of the objectness scoring module is that it gets as input a box B and yields two probabilities, p_{obj} and $p_{bck} = 1 - p_{obj}$, of whether or not this box tightly encloses an object, regardless of what the category of that object might be. The deep network architecture used for computing p_{obj} is presented in section 2.2.3.

2.2.3 AttractioNet architecture

We call the overall network architecture that implements the *Attend Refine Repeat* algorithm with its *In-Out* object location refinement module and its objectness scoring module, *AttractioNet*[1]. Given an image I, our *AttractioNet* model will be required to process multiple image boxes of various sizes, by two different modules and repeat those processing steps for several iterations of the *Attend Refine Repeat* algorithm. So, in order to have an efficient implementation we follow the SPP-Net [20] and Fast-RCNN [15] paradigm and share the operations of the first convolutional layers between all the boxes, as well as across the two modules and all the *Attend Refine Repeat* algorithm repetitions. Specifically, our *AttractioNet* model first forwards the image I through a first sequence of convolutional layers (conv. layers of VGG16-Net [37]) in order to extract convolutional feature maps F_I from the entire image.

[1] *AttractioNet : (Att)end (R)efine Repeat: (Act)ive Box Proposal Generation via (I)n-(O)ut Localization (Net)work*

Then, on each iteration t the box-wise part of the architecture gets as input the image convolutional feature maps F_I and a set of box locations $\mathbf{B}^{t-1}$ and yields the refined bounding box locations $\mathbf{B}^t$ and their objectness scores $\mathbf{O}^t$ using its object location refinement module sub-network and its objectness scoring module sub-network respectively. The architecture of its two sub-networks is described in more detail in the rest of this section:

Object location refinement module sub-network. This module gets as input the feature map F_I and the search region R and yields the probability vectors p_x and p_y of that search region with a network architecture similar to that of LocNet. Key elements of this architecture is that it branches into two heads, the X and Y, each responsible for yielding the p_x or the p_y outputs. Differently from the original LocNet architecture, the convolutional layers of this sub-network output 128 feature channels instead of 512, which speeds up the processing by a factor of 4 without affecting the category-agnostic localization accuracy. Also, in order to yield a fixed size feature for the R region, instead of region adaptive max-pooling this sub-network uses region bilinear pooling [9, 22] that in our initial experiments gave slightly better results. Finally, our version is designed to yield two probability vectors of size M[2], instead of $C \times 2$ vectors of size M (where C is the number of categories), since in our case we aim for category-agnostic object location refinement.

Objectness scoring module sub-network. Given the image feature maps F_I and the window B it first performs region adaptive max pooling of the features inside B that yields a fixed size feature ($7 \times 7 \times 512$). Then it forwards this feature through two linear+ReLU hidden layers of 4096 channels each (fc_6 and fc_7 layers of VGG16) and a final linear+softmax layer that yields two probabilities, p_{obj} and $p_{bck} = 1 - p_{obj}$, of whether or not box B tightly encloses an object. The fc_6 and fc_7 layers are followed by dropout units with dropout probability $p = 0.5$.

2.3 Training procedure

Training loss: During training the following multi-task loss is optimized:

$$\underbrace{\frac{1}{N^L}\sum_{k=1}^{N^L} L_{loc}(\theta|B_k,T_k,I_k)}_{\text{localization task loss}} + \underbrace{\frac{1}{N^O}\sum_{k=1}^{N^O} L_{obj}(\theta|B_k,y_k,I_k)}_{\text{objectness scoring task loss}}, \tag{1}$$

where θ are the learnable network parameters, $\{B_k,T_k,I_k\}_{k=1}^{N^L}$ are N^L training triplets for learning the localization task and $\{B_k,y_k,I_k\}_{k=1}^{N^O}$ are N^O training triplets for learning the objectness scoring task. Each training triple $\{B,T,I\}$ of the localization task includes the image I, the box B and the target localization probability vectors $T = \{T_x,T_y\}$. If (B_l^*,B_t^*) and (B_r^*,B_b^*) are the top-left and bottom-right coordinates of the target box B^* then the target probability vectors $T_x = \{T_{x,i}\}_{i=1}^M$ and $T_y = \{T_{y,i}\}_{i=1}^M$ are defined as:

$$T_{x,i} = \begin{cases} 1, & \text{if } B_l^* \le i \le B_r^* \\ 0, & \text{otherwise} \end{cases} \text{ and } T_{y,i} = \begin{cases} 1, & \text{if } B_t^* \le i \le B_b^* \\ 0, & \text{otherwise} \end{cases}, \forall i \in \{1,\ldots,M\} \tag{2}$$

The loss $L_{loc}(\theta|B,T,I)$ of this triplet is the sum of binary logistic regression losses:

$$\frac{1}{2M}\sum_{a\in\{x,y\}}\sum_{i=1}^{M} T_{a,i}\log(p_{a,i}) + (1-T_{a,i})\log(1-p_{a,i}), \tag{3}$$

[2]Here we use $M = 56$.

where p_a are the output probability vectors of the localization module for the image I and box B. The training triplet $\{B, y, I\}$ for the objectness scoring task includes the image I, the box B and the target value $y \in \{0, 1\}$ of whether the box B contains an object (positive triplet with $y = 1$) or not (negative triplet with $y = 0$). The loss $L_{obj}(\theta|B, y, I)$ of this triplet is the binary logistic regression loss $y\log(p_{obj}) + (1-y)\log(p_{bck})$, where p_{obj} and p_{bck} are the objectness probabilities for the image I and box B.

Creating training triplets: In order to create the localization and objectness training triplets of one image we first artificially create a pool of boxes that our iterative algorithm is likely to attend during test time. Hence we start by generating seed boxes (as the test time algorithm) and for each of them we predict the bounding boxes of the ground truth objects that are closest to them using an ideal object location refinement module. This step is repeated one more time using the previous ideal predictions as input. Because of the finite search area of the search region R the predicted boxes will not necessarily coincide with the ground truth bounding boxes. Furthermore, to account for prediction errors during test time, we repeat the above process by jittering this time the output probability vectors of the ideal location refinement module with 20% noise. Finally, we merge all the generated boxes (starting from the seed ones) to a single pool. Given this pool, the positive training boxes in the objectness localization task are those that their IoU with any ground truth object is at least 0.5 and the negative training boxes are those that their maximum IoU with any ground truth object is less than 0.4. For the localization task we use as training boxes those that their IoU with any ground truth object is at least 0.5.

Optimization: We use stochastic gradient descent (SGD) optimization with an image-centric strategy for sampling training triplets. Specifically, in each mini-batch we first sample 4 images and then for each image we sample 64 training triplets for the objectness scoring task (50% are positive and 50% are negative) and 32 training triplets for the localization task. The momentum is set to 0.9 and the learning schedule includes training for $320k$ iterations with a learning rate of $l_r = 0.001$ and then for another $260k$ iterations with $l_r = 0.0001$.

Scale and aspect ratio jittering: During test time our model is fed with a single image scaled such that its shortest dimension to be 1000 pixels or its longest dimension to not exceed the 1400 pixels. However, during training each image is randomly resized such that its shortest dimension to be one of the following number of pixels $\{300:50:1000\}$ (using Matlab notation) taking care, however, the longest dimension to not exceed 1000 pixels. Also, with probability 0.5 we jitter the aspect ratio of the image by altering the image dimensions from $W \times H$ to $(\alpha W) \times H$ or $W \times (\alpha H)$ where the value of α is uniformly sampled from $2^{.2:.2:1.0}$ (Matlab notation).

3 Experimental results

In this section we perform an exhaustive evaluation of our box proposal generation approach that we call *AttractioNet*. For that purpose, we train our model on the training set of MS COCO [27] that includes $80k$ images and we test it on the first 5k images of the COCO validation set and the PASCAL [10] VOC2007 test set (that also includes around $5k$ images).

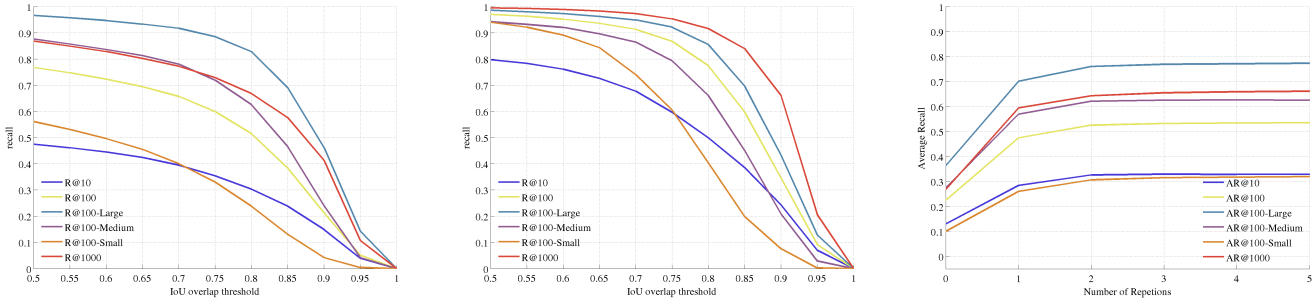
Evaluation Metrics: As evaluation metric we use the average recall (AR) which, for a fixed number of box proposals, averages the recall of the localized ground truth objects for several Intersection over Union (IoU) thresholds in the range .5:.05:.95 (Matlab notation). Specifically, we report the AR results for 10, 100 and 1000 box proposals using the notation *AR@10*, *AR@100* and *AR@1000* respectively. Also, in the case of 100 box proposals we also report the AR of the small ($\alpha < 32^2$), medium ($32^2 \leq \alpha \leq 96^2$) and large ($\alpha > 96^2$)

Method	AR@10	AR@100	AR@1000	AR@100-Small	AR@100-Medium	AR@100-Large
EdgeBoxes [43]	0.074	0.178	0.338	0.015	0.134	0.502
Geodesic [24]	0.040	0.180	0.359	-	-	-
Selective Search [38]	0.052	0.163	0.357	0.012	0.0132	0.466
MCG [2]	0.101	0.246	0.398	0.008	0.119	0.530
DeepMask [32]	0.153	0.313	0.446	-	-	-
DeepMaskZoom [32]	0.150	0.326	0.482	-	-	-
Co-Obj [19]	0.189	0.366	0.492	0.107	0.449	0.686
SharpMask [31]	0.192	0.362	0.483	0.060	0.510	0.665
SharpMaskZoom [31]	0.192	0.390	0.532	0.149	0.507	0.630
SharpMaskZoom^2 [31]	0.178	0.391	0.555	0.221	0.454	0.588
AttractioNet (Ours)	**0.328**	**0.535**	**0.661**	**0.319**	**0.625**	**0.773**
AttractioNet-PASCAL (Ours)	0.245	0.403	0.528	0.183	0.462	0.693

Table 1: Average Recall results on the first 5k images of COCO validation set.

Method	AR@10	AR@100	AR@1000	AR@100-Small	AR@100-Medium	AR@100-Large
EdgeBoxes [43]	0.203	0.407	0.601	0.035	0.159	0.559
Geodesic [24]	0.121	0.364	0.596	-	-	-
Selective Search [38]	0.085	0.347	0.618	0.017	0.134	0.364
MCG [2]	0.232	0.462	0.634	0.073	0.228	0.618
DeepMask [32]	0.337	0.561	0.690	-	-	-
Best of Co-Obj [19]	0.430	0.602	0.745	0.453	0.517	0.654
AttractioNet (Ours)	**0.554**	**0.741**	**0.851**	**0.562**	**0.670**	**0.788**

Table 2: Average Recall results on the PASCAL VOC2007 test set.



(a) COCO validation set **(b)** VOC2007 test set **(c)** COCO validation set

Figure 4: **(a)-(b)** Recall versus IoU plots of our approach for: 10 proposals (*R@10*), 100 proposals (*R@100*), 1000 proposals (*R@1000*), 100 proposals and small sized objects (*R@100-Small*), 100 proposals and medium sized objects (*R@100-Medium*) and 100 proposals and large sized objects (*R@100-Large*). **(c)** Average recall versus the repetitions number of the active proposal generation algorithm.

sized objects using the notation *AR@100-Small*, *AR@100-Medium* and *AR@100-Large* respectively, where α is the area of the object.

Implementation details: In the active box proposal algorithm we use 10k seed boxes generated with a similar to Cracking Bing [41] technique[3]. To reduce the computational cost of our algorithm, after the first repetition we only keep the top 2k scored boxes and we continue with this number of candidate box proposals for four more iterations. In the non-maximum-suppression [11] (NMS) step the optimal IoU threshold that is being used depends on the desired number of box-proposals. Specifically, for 10, 100 and 1000 proposals we use an IoU threshold of 0.55, 0.75 and 0.90 respectively (note that the aforementioned IoU thresholds were cross validated on a set different from the one used for evaluation).

3.1 Object box proposal generation evaluation

In Table 1 we report the average recall (AR) metrics of our method as well as of other competing methods in the COCO validation set. We observe that the average recall performance achieved by our method exceeds all the previous work in all the AR metrics by a significant margin (around 10 absolute points in the percentage scale). Similar gains are also observed in Table 2 where we report results in the VOC2007 test set. Furthermore, the first two plots

[3]We use seed boxes of 3 aspect ratios, $1:2$, $2:1$ and $1:1$, and 9 different sizes of the smallest seed box dimension $\{16, 32, 50, 72, 96, 128, 192, 256, 384\}$.

(plots (a) and (b)) of Figure 4 present in the case of our method the recall as a function of the IoU overlap of the ground truth objects. We see that the recall decreases relatively slowly as we increase the IoU from 0.5 to 0.75 while for IoU above 0.85 the decrease is faster.

In plot (c) of Figure 4 we provide the average recall metrics as a function of the repetitions number of our approach. We observe that the steepest increase in the AR metrics is when going from 0 repetitions (only objectness scoring of the seed boxes is performed) to 1 repetition (both objectness scoring and location refinement of the seed boxes). It is worth noting that after the 1st repetition, the AR performance of our approach is already better than the previous state-of-the-art (as reported in Table 1), which demonstrates the very good localization accuracy of our object location refinement module. Further increasing the repetitions number leads to an even higher AR performance, fact that validates our active box proposal generation strategy. Finally, it seems that the AR measurements start converging after the 4th repetition.

3.2 Generalization to unseen categories

In the final entry (*AttractioNet-PASCAL*) of Table 1 we report the average recall results on COCO validation set when our model is trained on PASCAL VOC07+12 train+val sets. The purpose of this experiment is to examine the ability of our approach to generalize to "unseen" categories since the 20 categories of the PASCAL dataset is only a small subset of the 80 categories of the COCO dataset). We observe that the average recall results of the *AttractioNet-PASCAL* entry are relatively close to those of the *AttractioNet* entry that is trained on the COCO training set. Given the fact that the *AttractioNet-PASCAL* entry is trained only on 16k images instead of 80k images as the *AttractioNet* entry, the performance difference between them is relatively small. By examining the AR metrics for the small, medium and large sized objects we observe that the main drop in performance is for the small and medium sized objects which we speculate is due to the fact that the COCO dataset mainly includes small and medium sized objects while the the PASCAL dataset is more biased towards the large sized objects. Furthermore, the *AttractioNet-PASCAL* entry is still better than the rest methods in Table 1 and especially for the *AR@10*, *AR@100* and *AR@100-Large* metrics it surpasses even the SharpMask entries that are the previous state-of-the-art in the field and are trained in the COCO training set. To conclude, we argue that the results of the *AttractioNet-PASCAL* entry prove experimentally the capability of our category agnostic location refinement model to generalize to "unseen" categories.

4 Conclusions

In our work we propose a bounding box proposals generation method, which we call *AttractioNet*, whose key elements are a strategy for actively searching of bounding boxes in the promising image areas and a powerful object location refinement module that extends the recently introduced LocNet [14] model on localizing objects agnostic to their category. We extensively evaluate our method on the challenging MS COCO and PASCAL datasets, demonstrating in both cases average recall results that surpass the previous state-of-the-art by a significant margin while also providing strong empirical evidence about the generalization ability of our approach w.r.t. unseen categories.

Acknowledgements

This work was supported by the *ANR SEMAPOLIS* project. We would like to thank Pedro O. Pinheiro for help with the experimental results and Sergey Zagoruyko for helpful discussions. We would also like to thank the authors of SharpMask [31] (Pedro O. Pinheiro, Tsung-Yi Lin, Ronan Collobert and Piotr Dollar) for providing us with its box proposals.

References

[1] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2012.

[2] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[3] Sean Bell, C Lawrence Zitnick, Kavita Bala, and Ross Girshick. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[4] Juan C Caicedo and Svetlana Lazebnik. Active object localization with deep reinforcement learning. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[5] Xiaozhi Chen, Huimin Ma, Xiang Wang, and Zhichen Zhao. Improving object proposals with multi-thresholding straddling expansion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[6] Ming-Ming Cheng, Ziming Zhang, Wen-Yan Lin, and Philip Torr. Bing: Binarized normed gradients for objectness estimation at 300fps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[7] Ramazan Gokberk Cinbis, Jakob Verbeek, and Cordelia Schmid. Weakly supervised object localization with multi-fold multiple instance learning. *arXiv preprint arXiv:1503.00949*, 2015.

[8] Jifeng Dai, Kaiming He, Yi Li, Shaoqing Ren, and Jian Sun. Instance-sensitive fully convolutional networks. *CoRR*, abs/1603.08678.

[9] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 2010.

[11] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2010.

[12] Amir Ghodrati, Ali Diba, Marco Pedersoli, Tinne Tuytelaars, and Luc Van Gool. Deepproposal: Hunting objects by cascading deep convolutional layers. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[13] Spyros Gidaris and Nikos Komodakis. Object detection via a multi-region & semantic segmentation-aware cnn model. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[14] Spyros Gidaris and Nikos Komodakis. Locnet: Improving localization accuracy for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[15] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[16] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, 2014.

[17] Abel Gonzalez-Garcia, Alexander Vezhnevets, and Vittorio Ferrari. An active search strategy for efficient object class detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[18] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*, 2015.

[19] Zeeshan Hayder, Xuming He, and Mathieu Salzmann. Learning to co-generate object proposals with a deep structured network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[20] K He, X Zhang, S Ren, and J Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2015.

[21] Jan Hosang, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele. What makes for effective detection proposals? *PAMI*, 2015.

[22] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[23] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[24] Philipp Krähenbühl and Vladlen Koltun. Geodesic object proposals. In *European Conference on Computer Vision*. Springer, 2014.

[25] Philipp Krähenbühl and Vladlen Koltun. Learning to propose objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[26] Weicheng Kuo, Bharath Hariharan, and Jitendra Malik. Deepbox: Learning objectness with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014.

[28] Yongxi Lu, Tara Javidi, and Svetlana Lazebnik. Adaptive object detection using adjacency and zoom prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[29] Francisco Massa, Bryan Russell, and Mathieu Aubry. Deep exemplar 2d-3d detection by adapting from real to rendered views. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[30] Mahyar Najibi, Mohammad Rastegari, and Larry S Davis. G-cnn: an iterative grid based object detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[31] Pedro H. O. Pinheiro, Tsung-Yi Lin, Ronan Collobert, and Piotr Dollár. Learning to refine object segments. 2016. URL http://arxiv.org/abs/1603.08695.

[32] Pedro O Pinheiro, Ronan Collobert, and Piotr Dollar. Learning to segment object candidates. In *Advances in Neural Information Processing Systems*, 2015.

[33] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2015.

[34] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[35] Kevin J Shih, Saurabh Singh, and Derek Hoiem. Where to look: Focus regions for visual question answering. 2016.

[36] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[37] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[38] Koen EA Van de Sande, Jasper RR Uijlings, Theo Gevers, and Arnold WM Smeulders. Segmentation as selective search for object recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 2011.

[39] Donggeun Yoo, Sunggyun Park, Joon-Young Lee, Anthony S Paek, and In So Kweon. Attentionnet: Aggregating weak directions for accurate object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[40] Sergey Zagoruyko, Adam Lerer, Tsung-Yi Lin, Pedro O Pinheiro, Sam Gross, Soumith Chintala, and Piotr Dollár. A multipath network for object detection. *arXiv preprint arXiv:1604.02135*, 2016.

[41] Qiyang Zhao, Zhibin Liu, and Baolin Yin. Cracking bing and beyond. In *Proceedings of the British Machine Vision Conference. BMVA Press*, 2014.

[42] Yukun Zhu, Raquel Urtasun, Ruslan Salakhutdinov, and Sanja Fidler. segdeepm: Exploiting segmentation and context in deep neural networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[43] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision*, 2014.