Feedback Convolutional Neural Networks for Visual Localization and Segmentation

Chunshui Cao, Yongzhen Huang, *Member, IEEE,* Yi Yang, Liang Wang, *Senior Member, IEEE,* and Tieniu Tan, *Fellow, IEEE,*

Abstract—Feedback, as a kind of very common and important mechanism in the human visual system, has not attracted sufficient attention in designing pattern recognition algorithms. In this paper, we propose that feedback plays a critical role for in-depth analysis of convolutional neural networks (CNN), e.g., how a neuron in CNN represents a part of an object, and how a set of local neurons form global perception of an object. To model the feedback in CNN, we develop a novel method consisting of two main operations, i.e., selective neural pathway pruning and pattern recovering, with mathematical analysis and strict proofs provided. We call the proposed model as feedback convolutional neural network (Feedback CNN), with which we can explicitly describe what a neuron represents with respect to a specific object category. The Feedback CNN model is easily trained with only category labels, but has the ability to localize and segment objects (like human learning), indicating that a classification model is capable of learning the essence of visual objects via well-designed feedback. The analysis of object/part visualization and relevant neuron selection reveals the potentially close relationship between neurons in Feedback CNN and parts of objects. Moreover, we design two quantitative experiments in terms of weakly supervised object localization and segmentation, respectively, and the experimental results on ImageNet and Pascal VOC show that our method largely outperforms the state-of-the-art ones.

Index Terms—Feedback, neural pathway pruning, neuron recovering, weakly supervised localization, weakly supervised segmentation.

1 Introduction

In recent years, convolutional neural networks (CNN) have made tremendous progress in a variety of vision tasks, including object classification [1–4], localization [5, 6], and semantic segmentation [7, 8]. These tasks are usually treated as different problems and handled separately with different frameworks. To localize and segment objects in images, most popular approaches [5–8] need strong supervision information for training, e.g., bounding boxes or segmentation masks. However, collecting a large amount of such data is often expensive and time-consuming. Therefore, it is ideal to localize and segment objects only when category labels are available, and this manner is naturally more like human learning.

Recently, the studies [9, 10] have shown that the convolutional units in CNNs only trained for the classification task have the potential to learn a part of semantic patterns like object parts or even the whole object. In these methods, category information is implicitly used as a kind of feedback signal in training. In neurobiology, the Biased Competition Theory [11] claims that when searching for objects in a scene, the human visual cortex is enhanced by the top-down stimuli after processing visual information in a bottom-up manner, and irrelevant neurons will be suppressed in the feedback loops, thus leading to the selectivity in neuron activations. This top-down feedback mechanism is involved in a variety of visual perception tasks, including selective attention, scene segmentation, and encoding and recalling learned information [12, 13].

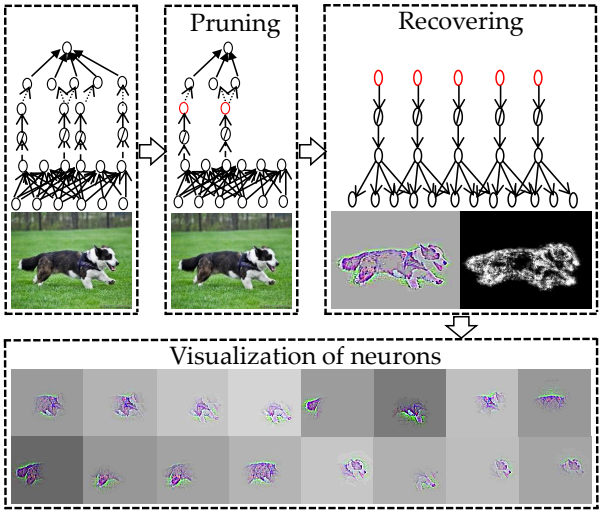
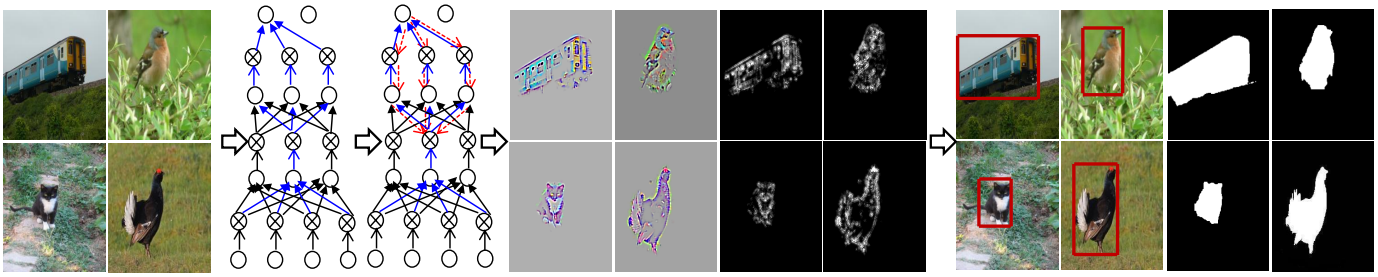


Fig. 1: Feedback CNN. Given an input image, we perform a normal feed-forward to predict the class label and set it as the target. Then use the pruning operation to select related neurons, and perform the recovering operation on these selected neurons to obtain target-relevant visualization and energy maps. Each selected neuron is highly related to the object parts, which is shown by visualizing the selected neurons respectively.

Inspired by the aforementioned observations, we propose a Feedback CNN to implement the unified processing of object recognition, localization and semantic segmentation. As shown

Chunshui Cao is with University of Science and Technology of China. Email: ccs@mail.ustc.edu.cn. Yongzhen Huang, Liang Wang and Tieniu Tan are with National Laboratory of Pattern Recognition, CAS Center for Excellence in Brain Science and Intelligence Technology, Institute of Automation, Chinese Academy of Sciences. Email: {yzhuang, wangliang, tnt}@nlpr.ia.ac.cn. Yi Yang is with Baidu Research. Email: yangyi05@baidu.com.



(a) Input images (b) Feed forward (c) Feedback (d) Visualization (e) Energy map (f) Localization (g) Segmentation

Fig. 2: A simple pipeline for object localization and segmentation via the proposed Feedback CNN model. Given an input image and a high-level semantic stimulus, we selectively prune the networks by back-propagating top-down information. After that, we can get target-relevant regions in visualization and energy maps, based on which object localization and segmentation can be easily achieved. Best viewed in color.

in Fig. 1, we first acquire target-relevant neurons by selectively pruning the neural pathway according to the predicted class label, and then simultaneously recover the activated patterns learned by the selected neurons in the image space. In this way, the object of interest can be effectively captured in visualization and energy maps. Specifically, we introduce **latent gate variables** to control the hidden neurons and formulate feedback as an optimization problem that is solved by our newly developed algorithms i.e., Feedback Selective Pruning (FSP) and Feedback Recovering (FR). FSP selects target-relevant neurons in hidden layers via back propagation pathways. FR restores visual information that falls in the receptive field of a given neuron. Consequently, combined with the advantages of both FSP and FR, the proposed Feedback CNN can effectively produce task-specific visualization and energy maps with high quality. In particular, we visualize several neurons selected by FSP in Fig. 1 via applying FR separately. The visualization results indicate that the selected neurons are highly relevant to the target object and correspond to different parts of the object. As a consequence, Feedback CNN selects target-relevant and suppresses irrelevant neurons during the top-down inference, guiding the model to focus on the most salient image regions that are highly related to the given category label.

Accordingly, Feedback CNN can force a normal CNN trained for object classification to localize and segment interesting objects from natural images, as illustrated in Fig. 2. More specifically, a CNN classifier takes an image in Fig. 2(a) as an input, then performs traditional feed-forward inference, as shown in Fig. 2(b). After that, the inferred category, e.g., "Train" for the first image, is set as the target for our feedback model in Fig. 2(c), and thus only the neurons related to "Train" will be activated. As a result, in Fig. 2(d)(e), only the salient regions related with "Train" are captured both in the visualization and the energy maps. With the help of these maps, it is easy to localize and even segment objects without any strongly supervised model learning, as shown in Fig. 2(f) and Fig. 2(g), respectively. As a consequence, Feedback CNN provides an innovative way to integrate object recognition, localization and segmentation into a unified framework, which is more like human learning.

This work makes a big progress along our previous one [14]. Most notably, the algorithms proposed in this paper are completely new, which are much more powerful for modeling feedback, based on which a new framework is developed. By a plenty of quantitative and qualitative analysis, we demonstrate that with these improvements the proposed Feedback CNN exhibits much stronger capability for task-specific neuron selection as well as capturing interesting objects. In particular, we apply this new framework for weakly supervised visual tasks for first time.

The main contributions of this paper are summarized as follows: 1) We develop two new algorithms (i.e., FSP and FR mentioned above) to model feedback mechanism in CNN as two effective solutions of the defined optimization problem with elegant mathematic proofs. 2) Via the visualization and energy maps, we demonstrate that the proposed Feedback CNN has the ability to select useful neurons related to interesting objects or its parts, as well as high signal-to-noise ratio, complete target description and clear object boundaries. 3) We apply Feedback CNN for weakly supervised object localization and segmentation, respectively. The big improvement over the state-of-the-art methods verifies the value of Feedback CNN.

2 Related Work

In this work, we will demonstrate that by selecting target-relevant hidden neurons and restoring the learned patterns of those selected neurons in the image space, a deep CNN merely trained for recognition can be generalized to localize and segment objects from images. Some studies closely related to this work are briefly discussed as follows.

2.1 Deep CNN

Recent years have witnessed the great success of deep CNN in various computer vision tasks [1–8]. Particularly, deep CNN has achieved human level performance for object recognition [4]. It learns features and classifiers simultaneously from a large scale of training samples [1–4, 15, 16]. The discriminative ability of deep CNN is greatly improved after a number of approaches being proposed, including dropout [17], PReLU [18], and batch normalization [19]. Moreover, there are considerable interests in enhancing deep CNN with greater capacity, in which the networks are generally designed to be deeper or wider [2, 4, 20]. The detailed analysis of CNN for the classification task [9, 10] shows that semantic patterns can be learned from the given training data.

All of these advances pave a way for constructing a feedback model in CNN. By introducing the feedback mechanism in those models, it is expected that we are capable of performing object localization and semantic segmentation easily under weakly supervised conditions.

2.2 Feedback Selection

Feedback selection plays an important role in human vision system, e.g., objects localization and segmentation from complex backgrounds [21], feature grouping [22], perceptual filling [23], and tuning neurons' receptive fields and functional roles [21]. Recently, some efforts have been made to embed feedback mechanism into deep neural networks. The convolutional latent variable models (CLVMs)in [24] model feedback as latent variables. An earlier study is presented in Deep Boltzmmann Machines (DBM) for feature selection [25]. Meanwhile, Recurrent Neural Networks (RNN) and Long Short Term Memory (LSTM) [26] are utilized to capture attention drifting in a dynamic environment and learn the feedback mechanism via reinforcement learning [27, 28]. On the other hand, Deep Boltzmmann Machines (DBM) [15, 29] and Deconvolutional Neural Networks [9] attempt to formulate feedback as a reconstruction process at the training stage.

In this paper, we propose to formulate feedback as an optimization problem for neuron selection. The main difference is that the proposed feedback is involved to selectively modulate the status of hidden neurons at the testing stage, and thus it does not affect the model training which is adopted in most previous studies introduced above. In particular, the feedback mechanism is embedded into a deep classification model, wherein feed-forward connections serve as information carriers, and useful information for a particular goal is selected by top-down feedback.

2.3 Weakly-supervised Object Localization and Segmentation

The direct applications of the proposed Feedback CNN are weakly supervised object localization and segmentation, and thus in this subsection we introduce some representatives in these two fields.

Many studies are developed for weakly supervised object localization using CNN [29–34]. To localize objects, a technique for self-teaching object localization is proposed in [32]. The approaches proposed in [29, 31] use global average pooling and max pooling to generate class-specific energy maps, and localize objects based on these maps. The work of [30] localizes objects by segmenting objects in an image based on the noisy energy map generated by class specified gradients. The approach in [33] needs re-training the recognition model with the average pooling layer. The method in [34] is based on a probabilistic "winner-take-all" process, in which activation values and positive convolutional weights are involved in the calculation of Marginal Winning Probability. The energy maps generated by [33] and [34] mainly highlight the most discriminative parts of objects but lose objects boundaries and suffer from noise and interference.

Recently, some remarkable approaches have been proposed for weakly-supervised semantic segmentation [35–39]. The approaches presented in [35] and [38] adopt different pooling strategies to train deep networks from the viewpoint of multiple instance learning. Both CCNN [37] and EM-Adapt [39] develop a self-training framework and enforce the consistency between the per-image annotation and the predicted segmentation masks by different constraints. Different from these methods, the proposed Feedback CNN in this paper can simultaneously perform object recognition, localization and semantic segmentation. Under the same weakly supervised settings, we merely need to train a classification model and then perform object localization and semantic segmentation based on the energy maps generated by the feedback selection mechanism. More details will be provided in the next section.

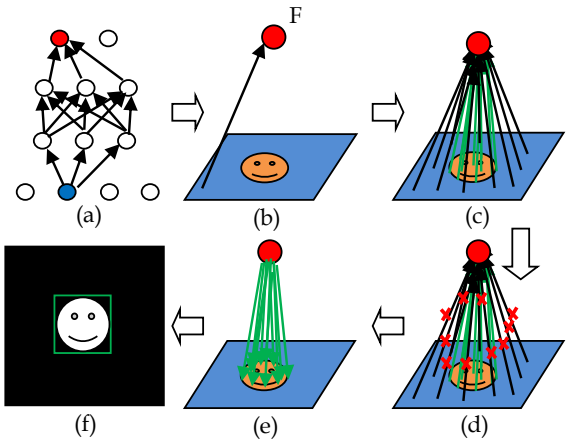


Fig. 3: Pathway selection. (a) Neural pathways between two neurons. (b) The neural pathways are abstracted as a CP. (c) All CPs between input pixels and a target neuron. (d) Selectively pruned CPs. (e) The preserved target-related CPs. (f) The results of object localization and segmentation.

3 Feedback CNN

3.1 Problem Definition and Formulation

3.1.1 Problem Description

For easy understanding, we use a figure to explain the meaning of the main variables in the definition. As shown in Fig. 3(a)(b), every pixel in the input image initiates several neural pathways to a high-level semantic neuron F which represents a face, and all these pathways can be abstracted as a connecting pathway (CP) between a pixel and F. Consequently, all pixels of the input image will be connected to F by their own CPs, as shown in Fig. 3(c). Let P denote the set of all these CPs. And all visual information of the face and the background is transmitted to the neuron F in a bottom-up manner. Let R be a rule to judge whether a CP links a target object pixel to the target neuron F or not. Then the set P can be divided into two subsets T and B according to R, as described in the following definition:

Problem definition: *Let* $P = \{$*CPs from all pixels to* $F\}$, $T = \{$*CPs from the target object to* $F\}$ *and* $B = \{$*CPs from the background to* $F\}$. *The problem is to find a rule* R, *s.t.* $P = T \bigcup B$ *and* $T \bigcap B = \emptyset$.

If we find the rule R and wipe out all the connections in B, as demonstrated in Fig. 3(d), the target object can be local-

ized and segmented in a top-down manner, which explains the technical feasibility of integrating object recognition, localization and segmentation together. To conclude, a strategy is required to selectively modulate the bottom-up pathways in neural networks according to the target semantic neuron.

In this paper, feedback serves as the rule R for the neural pathway selection. Specifically, we introduce latent gate variables and build feedback connections to control the hidden neurons, as demonstrated in Fig. 4. Neural pathways pruning is accordingly transferred to neuron selection. Fig. 4(c) demonstrates that feedback connections are established from top "goal" neurons to all latent gate variables. In addition to the bottom-up inference in traditional convolutional neural networks, Feedback CNN infers about the status of all the latent gate variables according to the "goal" of the network in several feedback loops. After the feedback selection, irrelevant hidden neurons will be suppressed which means that lots of non-target connecting pathways are clipped. Finally, we can obtain the information of objects in a top-down propagation.

3.1.2 Problem Formulation

CNNs usually consist of several stacked feed-forward layers, including the convolutional layer, the rectified linear units (ReLU) layer, the max pooling layer and some other nonlinear layers like the soft-max layer and the sigmoid layer. The ReLU and the max pooling layers can be interpreted as "gates" attached to the output of convolutional layers and the status of these "gates" are actually controlled by the input information. The network expresses patterns during the feed-forward phase in a bottom-up manner. For each output neuron in the convolutional layer, as long as the pattern lying on its receptive field is matched with its learned pattern and the similarity is stronger than its neighbors, the corresponding "gates" in the succedent ReLU and max pooling layers will be opened. When targeting at a particular semantic label, these activated neurons could be either helpful or harmful, and they potentially involve too much noise and irrelevant information. So it is reasonable to put additional gates on the output of the ReLU and max pooling layers to further turn off those activated but irrelevant neurons. In particular, these new gates are determined by both bottom-up feed-forward information and top-down feedback information.

We formulate such feedback mechanism as an optimization problem by introducing additional gate variables Z. Given an image I and a CNN with parameters W, we denote the score of the target neuron as S and the mapping function from I to S as $f(I)$. As mentioned above, the convolutional units are already controlled by ReLU and max pooling gates, thus we put additional gates Z on the top of each ReLU neuron and max pooling neuron.

Denote z_{ijc}^l as the additional gate for the neuron at location (i, j) and channel c in layer l, and denote the mapping function as $f(I, Z)$. The optimization problem is formulated as:

$$\begin{aligned} &\max_Z S = f(I, Z) \\ &s.t.\ \ z_{ijc}^l \in \{0, 1\}\ \forall\, l, i, j, c \\ &type(l)\ =\ ReLU\ or\ max\ pooling \end{aligned} \tag{1}$$

This leads to an integer programming problem, which is NP-hard with the current deep net architecture. Fortunately, locally optimal solutions can be derived for this problem when $f(I, Z)$ is linearly approximated.

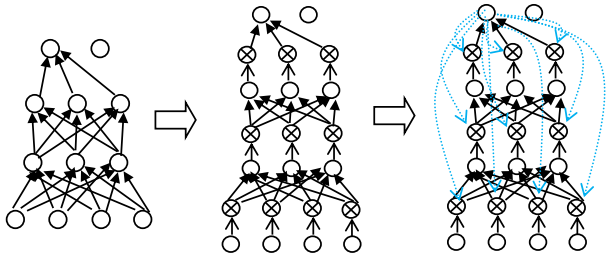


Fig. 4: Building the feedback connection. (a) An original CNN. (b) Put gates on each of the hidden neurons. (c) Build feedback connections between the target neuron and all the gate variables.

3.1.3 Linear Approximation

It is well known that CNN is a nonlinear mapping function, since it has some nonlinear layers such as the ReLU layer, the max pooling layer and so on. However, given an input image I_0, we can approximate the nonlinear CNN around I_0 in the input space with a linear function, denoted as $F(I_0)$. Specifically, for the neurons in ReLU and max pooling, we fix their status after the feed-forward procedure. And other neurons in the nonlinear mapping functions, e.g., $g(x)$, are approximated with their first order Taylor Expansion at x_0 which is a specific input determined by I_0, as described in Equation (2). Furthermore, since the max pooling layers always appear after the ReLU layers (like gating), we initialize their additional gates with their status after the first feed-forward and do not change them any more. That means the algorithm will concentrate on the additional gates for ReLU neurons.

$$\begin{aligned} &g(x) = g(x_0) + g'(x_0)(x - x_0) + o(x - x_0) \\ &g(x) \approx g(x_0) + g'(x_0)(x - x_0) \end{aligned} \tag{2}$$

After the above approximation operations, the target S can be described based on an ReLU layer:

$$\begin{aligned} &S = F(I_0, Z) = \textstyle\sum_{ijc}^l \alpha_{ijc}^l z_{ijc}^l x_{ijc}^l \\ &\alpha_{ijc}^l = \frac{\partial S}{\partial (z_{ijc}^l x_{ijc}^l)} \\ &type(l) = ReLU \end{aligned} \tag{3}$$

where x_{ijc}^l is the neuron at (i, j) of channel c in layer l. For convenience, layer l means the ReLU layer l in the rest of this paper. Note that we denote all the ReLU layers from bottom to top with index $l = 1, 2, \ldots, N$. And α_{ijc}^l can be calculated as the sum of weights of all neuron pathways from x_{ijc}^l to the target neuron S. Hence, we call α_{ijc}^l as the Summation Weight of pathways (SW). Then, the feedback optimization problem is transited as:

$$\begin{aligned} &\max_Z S = F(I_0, Z) \\ &s.t.\ \ z_{ijc}^l \in \{0, 1\}\ \forall\, l, i, j, c \\ &type(l)\ = ReLU \end{aligned} \tag{4}$$

Next, we present two solutions, and both of them are layer-by-layer optimizing procedure.

3.2 Solutions

3.2.1 Feedback Recovering

Due to the hierarchical network architecture of CNN, the target S in Equation (4) can be expanded via nested functions from top to bottom layer. Consequently, to maximize S, we can optimize additional gates Z layer-by-layer in a top-down order, as demonstrated in Algorithm 1. Note that we denote the mapping function of the target S after updating the ReLU layer l as S_l, and use subscript k to replace i, j, c for simplicity. $w_{k'}^{l-1}$ is the weight between $x_{k'}^{l-1}$ and x_k^l when the convolution operation is performed from layer $l-1$ to layer l. A sign function $\delta(x)$ is described as:

$$\delta(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases} \tag{5}$$

Algorithm 1 Feedback Recovering Algorithm

INPUT : $image\ I_0, target\ neuron\ with\ score\ function\ S$

DO :

$Initialize\ all\ Z\ with\ 1$

for $iteration = 1\ to\ max\ iteration$ **do**

$feedforward$

if $iteration == 1$ **then**

$Do\ linear\ approximation\ operations$

end if

for $l = N\ to\ 1$ **do**

if $l = N$ **then**

$\alpha_k^N = \frac{\partial S}{\partial x_k^N}$

$z_k^N = \delta(\alpha_k^N)$

update $\alpha_k'^N = z_k^N * \alpha_k^N$

update $S \to S_N = \sum_k^N \alpha_k'^N x_k^N$

else

Fix $z_k^N, z_k^{N-1}, \ldots, z_k^{l+1}$

$S_{l+1} = \sum_k^{l+1} \alpha_k'^{l+1} x_k^{l+1}$

$\alpha_k^l = \frac{\partial S_{l+1}}{\partial x_k^l}$

$z_k^l = \delta(\alpha_k^l)$

Update $\alpha_k'^l = z_k^l * \alpha_k^l$

update $S_{l+1} \to S_l = \sum_k^l \alpha_k'^l x_k^l$

end if

$l--$

end for

end for

To prove that the FR algorithm (Algorithm 1) can obtain local optimum, we need to prove that S will keep increasing after each iteration, which means that we should prove $S_N \leq S_1$. Accordingly, we first prove that $S_N \leq S_{N-1}$, then demonstrate that $S_l \leq S_{l-1}$ under the assumption of $S_{l+1} \leq S_l$, which following the rules of mathematical induction method.

1. When $l = N$:

Note that S can be described by any ReLU layer neuron.

$$S = F(I_0, Z) = \sum_k^N \alpha_k^N z_k^N x_k^N \tag{6}$$

x_k^N is an output of the ReLU neuron in layer N, so $x_k^N \geq 0$. Note that

$$\alpha_k^N z_k^N x_k^N \leq \alpha_k^N \delta(\alpha_k^N) x_k^N \tag{7}$$

Let $z_k^N \to z_k'^N = \delta(\alpha_k^N)$
and

$$\alpha_k'^N = \alpha_k^N * z_k'^N,\ where\ \alpha_k'^N \geq 0 \tag{8}$$

then

$$S \leq S_N = \sum_k^N \alpha_k'^N x_k^N \tag{9}$$

After updating all z_k^N in layer N, S_N can be expressed by layer $N-1$. Note that α_k^{N-1} is dependent on α_k^N, and it will be changed to $\hat{\alpha}_k^{N-1}$ when α_k^N being modified, then

$$S_N = \sum_k^{N-1} \hat{\alpha}_k^{N-1} z_k^{N-1} x_k^{N-1} \tag{10}$$

Update z_k^{N-1} and $\hat{\alpha}_k^{N-1}$ to get S_{N-1} in the same way when we update z_k^N and α_k^N, then

$$S_N \leq S_{N-1} \tag{11}$$

2. Let us assume that $S_{l+1} \leq S_l$:

Fix $z_k^N, z_k^{N-1}, \ldots, z_k^{l+1}$, then

$$S_l = \sum_k^l \alpha_k'^l x_k^l \tag{12}$$

Note that x_k^l can be expressed by $x_{k'}^{l-1}$ with convolutional weights $w_{k'}^{l-1}$:

$$x_k^l = relu(\sum_{k'}^{l-1} w_{k'}^{l-1} z_{k'}^{l-1} x_{k'}^{l-1}) \tag{13}$$

If $\sum_{k'}^{l-1} w_{k'}^{l-1} z_{k'}^{l-1} x_{k'}^{l-1} < 0$, there will be a zero term in S_l which can be ignored. So we just care about the case when $\sum_{k'}^{l-1} w_{k'}^{l-1} z_{k'}^{l-1} x_{k'}^{l-1} \geq 0$, then

$$x_k^l = \sum_{k'}^{l-1} w_{k'}^{l-1} z_{k'}^{l-1} x_{k'}^{l-1} \tag{14}$$

So,

$$S_l = \sum_k^l \alpha_k'^l \sum_{k'}^{l-1} w_{k'}^{l-1} z_{k'}^{l-1} x_{k'}^{l-1} \tag{15}$$

Note that $\alpha_k'^l \geq 0$, and then

$$S_l = \sum_{k'}^{l-1} (\sum_k^l \alpha_k'^l w_{k'}^{l-1}) z_{k'}^{l-1} x_{k'}^{l-1} \tag{16}$$

Next, update the gates of ReLU layer $l-1$ based on S_l.
Note that

$$\alpha_{k'}^{l-1} = \frac{\partial S_l}{\partial x_{k'}^{l-1}} = (\sum_k^l \alpha_k'^l w_{k'}^{l-1}) z_{k'}^{l-1} \tag{17}$$

Update $z_{k'}^{l-1} \to z_{k'}'^{l-1}$

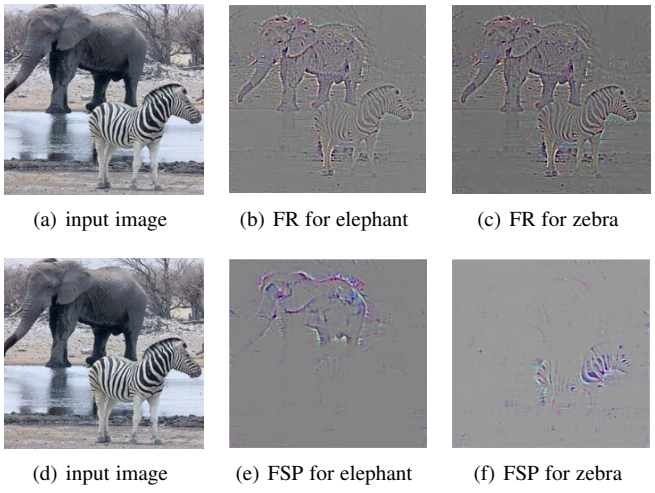
$$z_{k'}'^{l-1} = \delta(\frac{\partial S_l}{\partial x_{k'}^{l-1}}) \tag{18}$$

and $\alpha_{k'}^{l-1} \to \alpha_{k'}'^{l-1}$

$$\begin{aligned} \alpha_{k'}'^{l-1} &= (\sum_k^l \alpha_k'^l w_{k'}^{l-1}) * \delta(\sum_k^l \alpha_k'^l w_{k'}^{l-1}) \\ &= \frac{\partial S_l}{\partial x_{k'}^{l-1}} * \delta(\frac{\partial S_l}{\partial x_{k'}^{l-1}}) \\ &= (\sum_k^l \alpha_k'^l w_{k'}^{l-1}) * z_{k'}'^{l-1} \\ &\geq (\sum_k^l \alpha_k'^l w_{k'}^{l-1}) * z_{k'}^{l-1} \end{aligned} \tag{19}$$

note that $\alpha_{k'}'^{l-1} \geq 0$ and $x_{k'}^{l-1} \geq 0$, so

$$\begin{aligned} S_{l-1} &= \sum_{k'}^{l-1} \alpha_{k'}'^{l-1} x_{k'}^{l-1} \\ &\geq \sum_{k'}^{l-1} (\sum_k^l \alpha_k'^l w_{k'}^{l-1}) z_{k'}^{l-1} x_{k'}^{l-1} = S_l \end{aligned} \tag{20}$$



(a) input image (b) FR for elephant (c) FR for zebra

(d) input image (e) FSP for elephant (f) FSP for zebra

Fig. 5: Results generated by running the FR and the FSP algorithms for different targets respectively. (a) The input image for FR. (b)(c) The output maps generated by running FR for elephant and zebra respectively. (d) The same input image for FSP. (e)(f) The output maps generated by running FSP for elephant and zebra respectively. Best viewed in color.

That is

$$S_l \leq S_{l-1} \tag{21}$$

So based on the above mathematical induction, after the first iteration, the following conclusion can be drawn:

$$S_N \leq S_1 \tag{22}$$

The score S will keep increasing until convergence.

We take Fig. 5 as an example for explanation. As shown in the first row of Fig. 5, we embed the proposed FR algorithm into the VggNet [3] pre-trained on the ImageNet 2012 data set. Given the input image in Fig. 5(a) which contains an elephant and a zebra, we run FR for these two categories, respectively. After convergence, a back-propagation to the image space is performed to acquire the visualization maps. The results are depicted in Fig. 5(b) and (c). We find that the FR fails to distinguish particular patterns for different target objects, but can roughly restore the visual information lying on the receptive field of a target neuron, which is the reason that Algorithm 1 is named as the Feedback Recovering Algorithm. A major reason for these results is that we sequentially change SWs of hidden neurons throughout the optimization process. Note that SW indicates how much a hidden neuron contribute to the target neuron. More detailed analysis will be provided in the discussion section.

3.2.2 Feedback Selective Pruning

The fact that FR has weak discriminative ability by modulating SWs during its optimization reminds us to determine the additional gates Z by unchanged SWs. With this assumption, we modulate the input of each hidden neuron to maximize the target S. In particular, we optimize all the gates Z layer-by-layer in a bottom-up order as illustrated in Algorithm 2. To prove that the FSP algorithm (Algorithm 2) can also reach a local optimum, the mathematical induction method is adopted again. In this case, we need to prove $S_1 \leq S_N$. We first prove that $S_1 \leq S_2$,

Algorithm 2 The Feedback Selective Pruning Algorithm

INPUT : $image\ I_0, target\ neuron\ with\ score\ function\ S$

DO :

$Initialize\ all\ Z\ with\ 1$

for $iteration = 1\ to\ max\ iteration$ **do**

$feedforward$

if $iteration == 1$ **then**

$Do\ linear\ approximation\ operations$

end if

for $l = 1\ to\ N$ **do**

if $l = 1$ **then**

$\alpha_k^1 = \frac{\partial S}{\partial x_k^1}$

$z_k^1 = \delta(\alpha_k^1)$

update $x_k'^1 = z_k^1 * x_k^1$

update $S \rightarrow S_1 = \sum_k^1 \alpha_k^1 x_k'^1$

else

Fix $z_k^l, z_k^2, \ldots, z_k^{l-1}$

$S_{l-1} = \sum_{k'}^{l-1} \alpha_{k'}^{l-1} x_{k'}'^{l-1}$

In the other side,

$S_{l-1} = \sum_k^l \alpha_k^l x_k^l$

$x_k^l = relu(\sum_{k'}^{l-1} w_{k'}^{l-1} z_{k'}^{l-1} x_{k'}'^{l-1})$

$\alpha_k^l = \frac{\partial S_{l-1}}{\partial x_k^l}$

$z_k^l = \delta(\alpha_k^l)$

Update $x_k'^l = z_k^l * x_k^l$

update $S_{l-1} \rightarrow S_l = \sum_k^l \alpha_k^l x_k'^l$

end if

$l++$

end for

end for

and then illustrate $S_l \leq S_{l+1}$ under the assumption of $S_{l-1} \leq S_l$.

1. When $l = 1$:
Note that

$$S = F(I_0, Z) = \sum_k^1 \alpha_k^1 z_k^1 x_k^1 \tag{23}$$

x_k^1 is an output of an ReLU neuron in layer 1, so $x_k^1 \geq 0$.
Note that

$$\alpha_k^1 z_k^1 x_k^1 \leq \alpha_k^1 \delta(\alpha_k^1) x_k^1 \tag{24}$$

Let $z_k^1 \rightarrow z_k'^1 = \delta(\alpha_k^1)$, then

$$x_k'^1 = x_k^1 * \delta(\alpha_k^1),\ note\ that\ x_k'^1 \geq 0 \tag{25}$$

and thus

$$S \leq S_1 = \sum_k^1 \alpha_k^1 x_k'^1 \tag{26}$$

After update all z_k^1 in layer 1, S_1 can be expressed by layer 2. Note that x_k^2 will be changed to $\hat{x}_k^2$ because of x_k^1 being modified, that is

$$S_1 = \sum_k^2 \alpha_k^2 z_k^2 \hat{x}_k^2 \tag{27}$$

Update z_k^2 and $\hat{x}_k^2$ to get S_2 with the same way when we update z_k^1 and x_k^1, therefore:

$$S_1 \leq S_2 \tag{28}$$

2. Let us assume that $S_{l-1} \leq S_l$:
Fix $z_k^1, z_k^2, \ldots, z_k^{l-1}$, and then

$$S_l = \sum_k^l \alpha_k^l x_k'^l \tag{29}$$

The score S can be expressed by x_k^{l+1}, so

$$S_l = \sum_k^{l+1} \alpha_k^{l+1} \hat{x}_k^{l+1} z_k^{l+1} \tag{30}$$

where

$$\hat{x}_k^{l+1} = relu(\sum_{k'}^l w_{k'}^l z_{k'}^l x_{k'}'^l) \tag{31}$$

and thus $\hat{x}_k^{l+1} \geq 0$
Because

$$\begin{aligned} S_l &= \sum_k^{l+1} \alpha_k^{l+1} \hat{x}_k^{l+1} z_k^{l+1} \\ &\leq \sum_k^{l+1} \alpha_k^{l+1} \delta(\alpha_k^{l+1}) \hat{x}_k^{l+1} \\ &= \sum_k^{l+1} \alpha_k^{l+1} x_k'^{l+1} = S_{l+1} \end{aligned} \tag{32}$$

that is, $S_l \leq S_{l+1}$
Update $\hat{x}_k^{l+1} \rightarrow x_k'^{l+1}$ and $z_k^{l+1} \rightarrow z_k'^{l+1}$ with

$$\begin{aligned} x_k'^{l+1} &= \hat{x}_k^{l+1} \delta(\alpha_k^{l+1}) \\ z_k'^{l+1} &= \delta(\alpha_k^{l+1}) \end{aligned} \tag{33}$$

Based on the above mathematical induction methods, after the first iteration, we have

$$S_1 \leq S_N \tag{34}$$

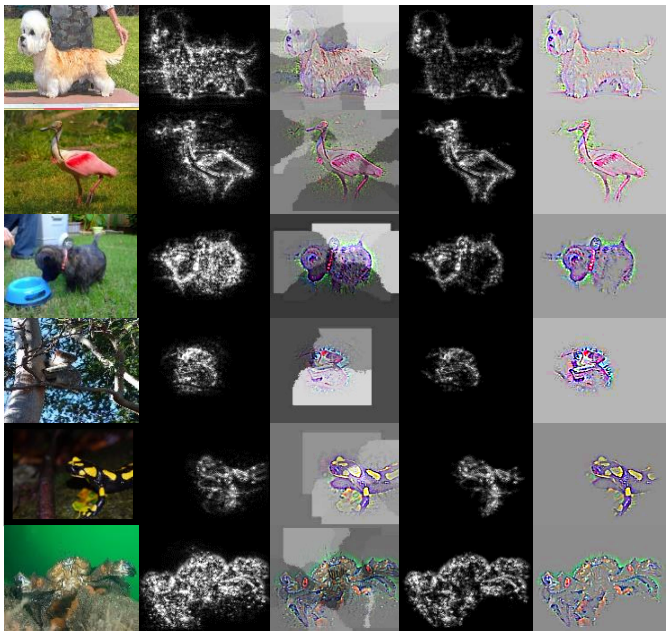
The target S will keep increasing until convergence.

As shown in Fig. 5, we apply the FSP for elephant and zebra using the same input image and VggNet, respectively. Fig. 5(e) and (f) illustrate the results. It is obvious that compared with the FR algorithm, the FSP algorithm has much more discriminative ability between different target objects. The salient regions in Fig. 5(e) and Fig. 5(f) are different targets separately. Hence, the FSP algorithm has the capacity to select neurons in deep CNN according to a pre-defined label, and that is why we call this algorithm as Feedback Selective Pruning. The main reason for these results is that the states of gate variables are determined by the SWs of hidden neurons and the inputs are modified instead of the SWs during the optimization process. We will further discuss this in the next section.

3.3 Discussion

As mentioned before, hidden neurons in a deep CNN are represented for particular patterns in the image space [9, 10, 30]. Also we show, in Equation (3), that a hidden neuron denoted by x_{ijc}^l can represent a particular pattern which is expressed by the weights of pathways between the related input pixels and this neuron. In particular, the strength of x_{ijc}^l is the similarity measurement when given an input pattern, and SW, namely α_{ijc}^l, reflects how this input pattern contributes to the target neuron S.

To maximize the target score S, the FR algorithm updates SWs layer-by-layer from the top to the bottom. During optimization, the additional gate z_{ijc}^l for x_{ijc}^l is determined by the modified $\hat{\alpha}_{ijc}^l$ via $\delta(\hat{\alpha}_{ijc}^l)$ instead of original α_{ijc}^l. As a consequence, updating α_{ijc}^l destroys the discriminative ability of the target neuron to judge whether a pattern is beneficial to semantic information it represents. But interestingly, the FR algorithm provides a good method to visualize the content in the receptive field of a neuron.



(a) (b) (c) (d) (e)

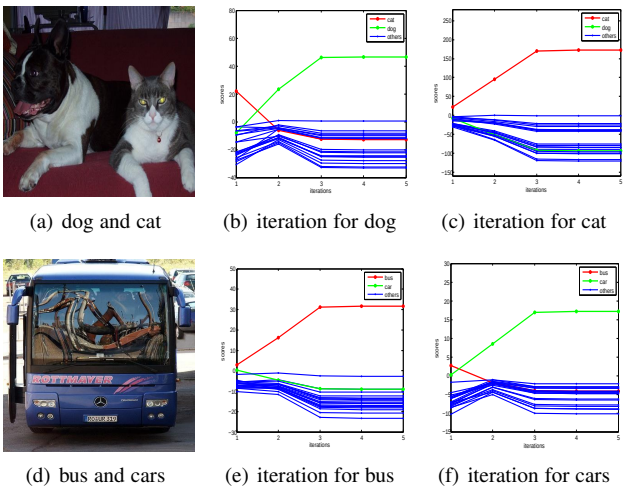
Fig. 6: More results generated by Feedback CNN. (a) Input images. (b) and (c) Merged energy maps and visualization maps by running FR separately on neurons selected by FSP. (d) and (e) Energy and visualization maps by running FR simultaneously on neurons selected by FSP. Best viewed in color.

In contrast, the FSP algorithm updates the activations of hidden neurons to maximize the target score S from the bottom to the top layer. For a particular neuron x_{ijc}^l, its gate status is determined by $\delta(\alpha_{ijc}^l)$, although the value of x_{ijc}^l may have changed during updating x_{ijc}^{l-1}. Moreover, S can be reinterpreted in the following way:

$$S = \frac{1}{N} \sum_{l=1}^{N} \sum_{ijc}^l \alpha_{ijc}^l x_{ijc}^l \tag{35}$$

If all $x_{ijc}^l \ \forall \ c, l \in 1, 2, \ldots, N$ represent different patterns, then S is a linear combination of all those patterns, as demonstrated in Equation (35). All the patterns with the negative SW will be erased by FSP. This will change the values of the preserved x_{ijc}^l, but it will not change the relationship between the preserved patterns and the target neuron. The FSP offers a good way to seek patterns closely related to particular target objects.

Because neither FR nor FSP can provide a global optimum solution, it is difficult to obtain perfect visualization and energy maps only by either of them. Fortunately, combined with the advantages of both, the proposed Feedback CNN can produce impressive results. In Feedback CNN, the FSP algorithm is utilized to select target-relevant neurons, and the FR algorithm is employed to reconstruct the target object in the image space. Fig. 6 presents more results. Specially, Fig. 6(b) and (c) are generated by running FR on selected neurons separately and merging the energy and the visualization maps together. Fig. 6(d) and (e) are obtained by



(a) dog and cat (b) iteration for dog (c) iteration for cat

(d) bus and cars (e) iteration for bus (f) iteration for cars

Fig. 7: The iteration curve of FSP for different objects. (a)(d) Input images containing multiple objects. (b)(c) The iteration curves for dog and cat respectively with the input of (a). (e)(f) The iteration curve for bus and car respectively with the input of (d). Best viewed in color.

running FR on selected neurons simultaneously which is much more efficient. As can be seen, objects are captured by Feedback CNN even when they are hidden in cluttered backgrounds, and neurons related with the target objects will be selected but irrelevant neurons will be turned off.

4 Experiments

In this section, extensive experiments are carefully designed to verify the effectiveness of the proposed Feedback CNN. The iteration process of FSP is analyzed in Section 4.1 and the effectiveness of neuron selection is studied in Section 4.2. We evaluate the discriminative ability of FSP in Section 4.3. Besides, we conduct quantitative experiments of weakly supervised object localization in Section 4.4 and weakly supervised semantic segmentation in Section 4.5. It should be noted that since the FR algorithm is like a kind of image reconstruction, we evaluate FR together with FSP in Section 4.4 and Section 4.5.

4.1 Analysis on Iteration Process of FSP

In order to verify our theoretical analysis described in Section 3 that the score of the target neuron would keep increasing until convergence when running the FSP algorithm, we visualize the iterative process of the FSP algorithm here. For experimental purposes, the VggNet (16 layers) [3], which is obtained from Caffe [40] model zoo and pre-trained with ImageNet 2012 training set, is fine-tuned on the Pascal VOC2012 data set.

As shown in Fig. 7(a), (b) and (c), given the input image, the FSP algorithm is applied respectively on two neurons which represent the categories of "dog" and "cat" in the last fully connected layer named as "fc8" in the VggNet. The scores of all the 20 neurons in "fc8", corresponding to the 20 classes of Pascal VOC2012, are recorded during the iteration procedure. The iteration curve for category "cat" is plotted with the red line, "dog" with the green line, and other 18 classes with the blue lines. As can be seen, all the iterative procedure converge after about 5 iterations. And the scores of the target neuron keep increasing until convergence, while the scores of other classes are suppressed even if the corresponding objects are presented in the image. The same results happen when given an image contains a bus and several cars, as shown in Fig. 7(d), (e) and (f). These results prove that FSP will converge to a local optimum efficiently, and increase the score of the target neuron effectively.

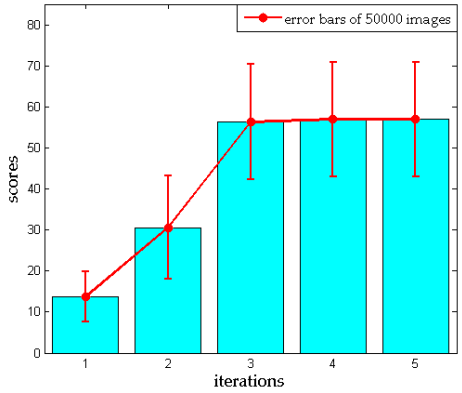


Fig. 8: The mean iteration curve of 50000 images from the ImageNet 2012 classification validation set.

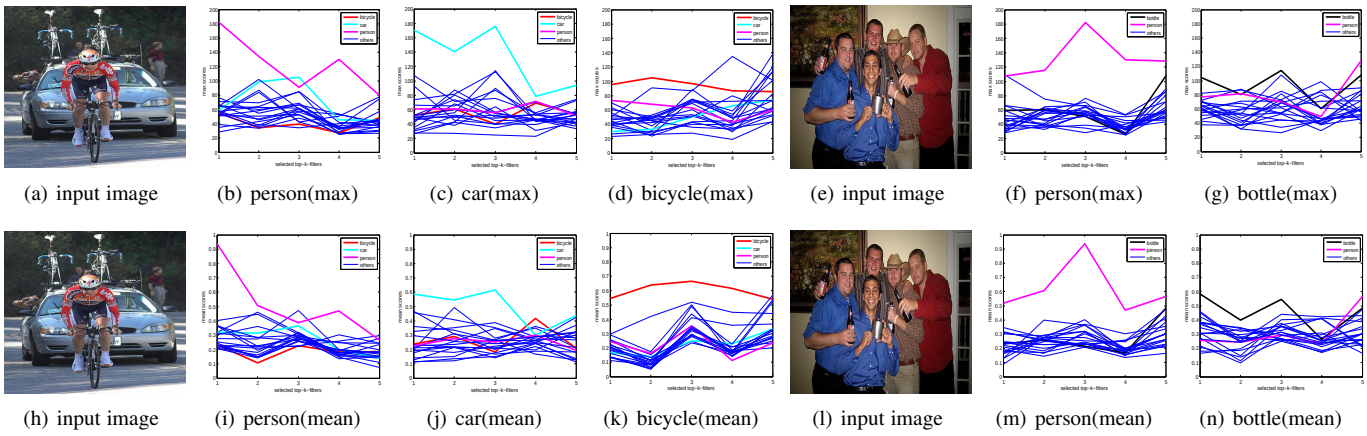
Furthermore, the FSP algorithm is applied on the ImageNet 2012 classification validation set which contains 50000 images. The ground-truth label of each image is set as the target for the feedback model, and the scores of 5 iterations for all images are recorded. Then we calculate their mean and standard deviation score of each iteration, and plot them in Fig. 8. We find that the FSP algorithm is also effective even for a very large image data set.

In addition, it should be noted that there are several small cars in the top left and right corners in Fig. 7(d). When feedback is applied with respect to category "car", the scores of the target neuron keep increasing while the scores for "bus" decrease heavily although there is a big bus in the center of the image, as demonstrated in Fig. 7(f). The reason is that neurons carrying useful information for particular targets can be selected effectively while irrelevant neurons will be turned off in the feedback loops. More analysis will be presented in the next experiment.

4.2 Effectiveness of Neuron Selection

In cognitive science, visual attention in the Biased Competition Theory [11, 41, 42] is explained as that human visual cortex is enhanced by top-down stimuli, and irrelevant neurons will be suppressed in feedback loops when searching for objects. When applying the FSP algorithm to the CNN model, we find that the results are similar to that described above.

Given an image with multiple classes, we run the FSP algorithm with the same VggNet in Section 4.1 for different targets, and focus on a middle layer named "conv5_2" which lies in the 12th layer of the total 16 layers. This layer has 512 filter kernels, indicating that it may express 512 patterns related to different classes.



(a) input image (b) person(max) (c) car(max) (d) bicycle(max) (e) input image (f) person(max) (g) bottle(max)

(h) input image (i) person(mean) (j) car(mean) (k) bicycle(mean) (l) input image (m) person(mean) (n) bottle(mean)

Fig. 9: Filter selection. Given input images like (a) and (e), the FSP algorithm is run for different objects in these images, and we select top 5 different channels for each target object according to the maximum scores of 512 channels in the "conv5_2" layer. Those 5 channels correspond to 5 filters. We feed the images of 20 different classes of Pascal VOC2012 validation set to CNN, and calculate the maximum and mean scores of those 5 filters corresponding to the images of different classes. The first row reports the maximum scores curve and the second row reports the mean scores curve. The filters selected by FSP well respond to the corresponding class images. For example, the scores of selected filters for cars are much higher than other classes when feeding the images of cars to CNN. Best viewed in color.

Fig. 10: Visualization of 5 neurons that have the maximum scores in each of the top 5 channels selected by the FSP algorithm. Those 5 neurons represent for the most discriminative parts of the corresponding target objects in the input image. Best viewed in color.

As shown in Fig. 10, the first input image contains persons, bicycles and a car, and the FSP algorithm is run for the three classes. After convergence, for each target, we rank the 512 channels of the "conv5_2" layer with their maximum scores and acquire the top 5 channels. Further, 5 neurons are selected that have the maximum activation scores in each of the top 5 channels. As illustrated in Fig. 10, the FR algorithm is run to visualize those 5 neurons selected by FSP, and it turns out that they represent the most discriminative parts of the corresponding objects. Similar results appear in another image with category person and bottle.

To make it more convincing, we evaluate these selected top 5 filters on the whole Pascal VOC2012 classification validation set which contains 1449 images. To avoid the underlying mutual influence, only 924 images with a single label are utilized, and these images are divided into 20 sets according to their labels. We calculate the maximum and mean responses of the top 5 channels with all images of 20 categories. In Fig. 9, the maximum and mean responses are presented in the first row and the second row respectively. We take the category "person" as an example for detailed analysis. FSP is run for the category "person" on a person-bike-car image until convergence, and top 5 channels are acquired. All images that are only labeled as person are fed to the original CNN model. The responses are drawn with the magenta line. Then, images of other 19 classes are fed to the same CNN to get the corresponding responses of the top 5 channels. The responses for the category "bicycle" and "car" that appear in the image are plotted with the red and cyan line respectively, and the rest 17 classes are plotted with blue lines.

In Fig. 9(b), the fact that the magenta line is higher than other lines indicates that the corresponding channels are highly related to its target category, and the FSP algorithm can effectively select the meaningful filters. The results are similar for another image that contains persons and bottles, as shown in Fig. 9(e). We find that this kind of neuron selection happens in all hidden layers, based on which we can draw a conclusion that the FSP algorithm has the ability to correctly select the corresponding neurons (filters) to preset targets, as well as suppress irrelevant neurons at the same time.

4.3 Analysis on the Discriminative Ability of FSP

To evaluate the discriminative ability of FSP, we conduct several experiments on the Pascal VOC2012 classification validation set. The same VggNet in Sec 4.1 is employed, and Feedback CNN is utilized to generate category-specific energy maps.

As a result, Fig. 11 depicts several examples. The energy maps generated by Feedback CNN are highly relevant to the target objects in the input images, as shown in Fig. 11(b), (f), (j) and

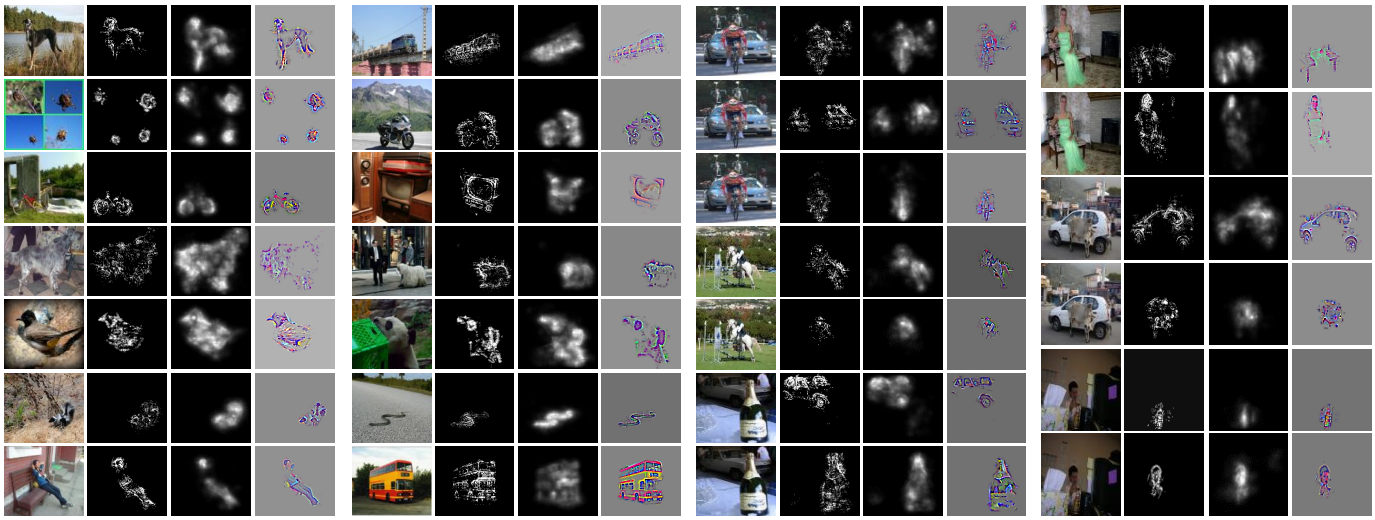


Fig. 11: Visualization and energy maps. (a) Input images. (b) FSP-FR energy maps. (c) Summation Energy Maps. (d) Visualization maps. (i-p) Some results when input images contain multi-class objects. Note that Summation Energy Maps are generated by summation of all resized gradients of feature maps in all convolutional layers. Best viewed in color.

(n). For convenience, these energy maps are named as FSP-FR energy maps.

Due to the selection ability of the FSP algorithm, most of the neurons preserved in all hidden layers are highly relevant to the same semantic class. Meanwhile, a whole object can be divided into several parts which may be expressed in several different hidden layers. Thus, it is reasonable to combine all the selected neurons to generate an new energy map. We achieve this simply by summation operations. After applying the FSP algorithm, we resize the gradients of feature maps in all convolutional layers of VggNet with the same size of the input image, and calculate the summation of all the resized gradient maps along the channel direction. The summation map is normalized by L2 norm, named as Summation Energy Map. Fig. 11(c), (g), (k) and (o) illustrate some results. As can be seen, the Summation Energy Maps have a better distributions over target objects.

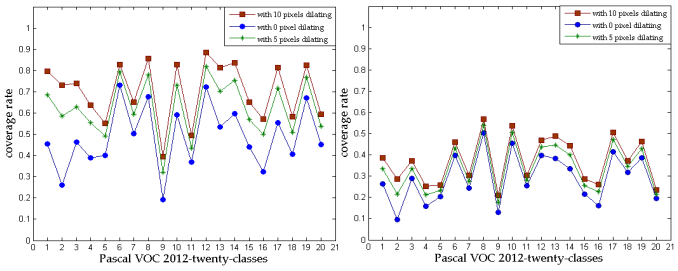


Fig. 12: Coverage rate of Summation Energy Map over all 20 classes images of Pascal VOC2012. All coverage rates of Summation Energy Maps (left) is much higher than the energy maps generated by original gradients of the input images (right). Best viewed in color.

The Summation Energy Map integrates the activation values of the selected neurons in all hidden layers. So it is more convincing that we evaluate the discriminative ability of FSP using Summation Energy Maps instead of FSP-FR energy maps. We calculate the Summation Energy Maps for each image of the Pascal VOC2012 segmentation validation set. As the data set provides ground-truth masks for each object in every image according to class labels, we calculate the sum of energy that falls into the target object regions, and we call this value as the coverage rate. The mean coverage rate is computed for each class on all provided validation images in Fig. 12. Specially, since the deconvolutional operation in back-propagation causes dilation of the edges in the energy maps, we further report the results of dilating the ground truth masks by 5 pixels and 10 pixels in Fig. 12. As a contrast, the mean coverage rate of energy maps which depends on the original gradients of the input image is reported too. As can be seen, the Summation Energy Maps generated by FSP effectively highlight the expected objects and focus on the target area.

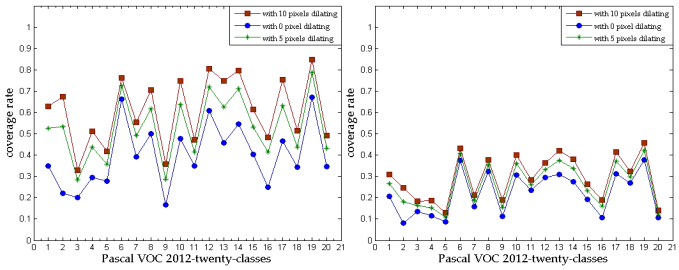


Fig. 13: Coverage rate of Summation Energy Map of multi-class images of Pascal VOC2012. All coverage rates of Summation Energy maps (left) is much higher than the energy maps generated by original gradients of the input images (right). Best viewed in color.

Particularly, to provide a more convincing evaluation of the discriminative ability of Summation Energy Map, we calculate the coverage rate only for images with multi-class labels in PASCAL VOC2012 segmentation validation set, which has 257 images. The corresponding results are shown in Fig. 13, demonstrating the effectiveness of FSP.

These results indicate that the propose FSP algorithm has

strong discriminative ability. Object-related neurons can be correctly selected by FSP and class-specific energy maps can also be effectively produced, which well paves the road for weakly supervised object localization and weakly-supervised semantic segmentation.

4.4 Weakly-supervised Object Localization

In this section, we evaluate the localization power of Feedback CNN on the ImageNet 2012 localization task. The top 5 localization evaluation metric [30] is employed, in which an correct prediction is counted when one of the top 5 guesses meets the requirement that both object category prediction and its associated bounding box are correct. To generate top 5 category prediction, the VggNet pre-trained with the ImageNet 2012 classification training set is downloaded from Caffe [40] model zoo. For weakly supervised localization, several steps are performed to get a bounding box. We summarize the experimental procedures of weakly supervised object localization in Algorithm 3.

Algorithm 3 Experimental procedures of weakly supervised object localization

1: Given an image and a predicted label;
2: Subtract the mean values and resize the image to the size of 224*224 as the input;
3: Run FSP according to the predicted label and obtain Summation Energy Map;
4: Get a bounding box which preserves 99% energy of the Summation Energy Map. Crop the box region from the original image and resize to 224*224 as the input;
5: Apply FSP again on the new input image;
6: Set one of the middle-level layers (e.g., "conv5_2") as the target layer for FR;
7: Set the preserved neurons in "conv5_2" as the targets, and run FR on those neurons simultaneously to get the final energy maps;
8: Get a bounding box which preserves 99% energy of the final energy map.

TABLE 1: Localization results on ILSVRC2012.

methods	top 5 classification error(%)	top 5 localization error(%)
deepinside [30]	-	44.6
VGGnet-GAP [33]	12.2	45.14
Backprop-on-VGGnet [33]	11.4	51.46
GoogLeNet-GAP [33]	13.2	43.00
GoogLeNet [33]	11.3	49.34
Feedback CNN-no crop	**15.68**	**42.82**
Feedback CNN-5 crop	**12.95**	**41.72**
Feedback CNN-dense crop	**9.22**	**40.32**
MWP [34]	with GT	38.70
Feedback CNN	with GT	**36.50**

As described in Algorithm 3, objects are localized in two stages for the reason that the scale of objects may vary greatly between different input images. We first roughly localize the objects in Step 1-4 by running FSP. Then the precise localization of objects is obtained in Step 5-9 by combining FSP and FR.

We compare the localization performance of Feedback CNN on the ILSVRC2012 validation set with several state-of-the-art methods in Table 1. Different image cropping strategies, such as no cropping, 5 cropping, and dense cropping [3], are employed to produce different classification accuracy. When both using VggNet, compared with the VGGnet-GAP in [33], our method wins 5.01% in terms of the accuracy of weakly supervised object localization. To avoid the influence of different classification performance of the compared models (the correctness of feedback signals), we compare the localization accuracy when classification error rates are very close. As illustrated in Table 1, when our classification accuracy is 3.48% (without cropping operation) and 0.75% (with 5 cropping operations) lower than the compared approach VGGnet-GAP [33], we still achieve 2.32% and 3.42% higher localization accuracy, respectively. Moreover, when given ground truth labels, 36.50% error rate is obtained, which is an accuracy of 2.2% higher than the recent best-performing approach MWP [34] under the same experimental set-up.

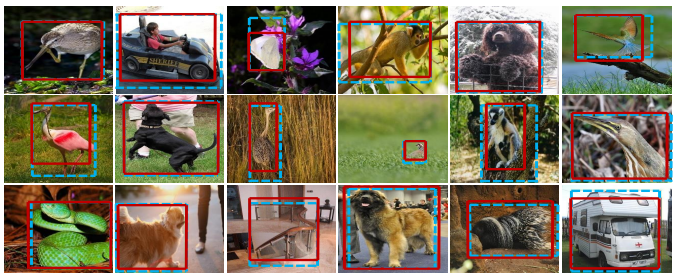


Fig. 14: Examples of weakly supervised localization of our approach. Note that red bounding boxes are predicted as one.

Due to the powerful selection capability of FSP and the better object boundaries in energy maps, the proposed Feedback CNN outperforms the compared approaches (VGGnet-GAP [33] and MWP [34]). The energy maps generated by our Feedback CNN are more precise and contain more complete objects. Accordingly, the bounding boxes generated by our approach are more close to the ground-truth bounding boxes. Fig. 14 displays some examples.

4.5 Weakly-supervised Semantic Segmentation

In this section, we focus on the weakly-supervised semantic segmentation task with experimental analysis on the Pascal VOC2012 semantic segmentation Challenge. We employ the standard Pascal VOC2012 segmentation metric: mean intersection-over-union (mIoU). Note that we only make use of class-level labels to fine-tune VggNet for classification on the Pascal VOC2012 segmentation training set, and evaluate our method on the Pascal VOC2012 semantic segmentation validation set (containing 1449 images). In the training phase, the input images are randomly cropped, mirrored, scaled and rotated to obtain a better model. To segment objects from an input image based on the energy map, the saliency cut proposed in [43] is utilized.

Algorithm 4 describes the experimental procedure. In particular, distinct parts of an object may be expressed in different layers, and their information can be all integrated into the Summation Energy Maps, which makes the Summation Energy Maps suitable for the segmentation task. On the other hand, the FSP-FR energy maps can highlight object boundaries. Thus, we acquire both these energy maps for the target objects in an input image and simply add them together as the final energy map, which is called

TABLE 2: Results of weakly supervised semantic segmentation on the Pascal Voc2012 validation dataset.

	bkg	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mAP
img+obj [35]																						32.2
stage1 [36]	71.7	30.7	**30.5**	26.3	20.0	24.2	39.2	33.7	50.2	17.1	29.7	22.5	41.3	35.7	43.0	36.0	29.0	34.9	23.1	33.2	33.2	33.6
EM-Adapt [39]	67.2	29.2	17.6	28.6	22.2	29.6	47.0	44.0	44.2	14.6	35.1	**24.9**	41.0	34.8	41.6	32.1	24.8	37.4	24.0	38.1	31.6	33.8
CCNN [37]	68.5	25.5	18.0	25.4	20.2	36.3	46.8	47.1	48.0	15.8	37.9	21.0	44.5	34.5	46.2	**40.7**	**30.4**	36.3	22.2	38.8	36.9	35.3
MIL+ILP+SP [38]	77.2	37.3	18.4	25.4	28.2	31.9	41.6	48.1	50.7	12.7	45.7	14.6	50.9	44.1	39.2	37.9	28.3	44.0	19.6	37.6	35.0	36.6
ours	**81.1**	**62.1**	25.9	**51.5**	**32.5**	**47.7**	**57.7**	**51.0**	**65.1**	**20.6**	**55.6**	23.7	**54.5**	**54.6**	**57.3**	38.5	27.2	**65.9**	31.2	**50.7**	**40.3**	**47.4**
classification acc		91.6	86.9	86.8	85.4	71.7	83.1	78.8	91.0	70.4	83.6	78.6	83.3	86.9	88.9	87.9	68.3	83.3	71.7	96.5	77.9	82.6

Fig. 15: Examples of weakly supervised semantic segmentation on the Pascal VOC2012 validation set. The first row are input images, the second row are ground truth, and the last row are our results.

Summation-FSP-FR energy map. For the overlapped objects, the pixels of the overlapped regions are simply determined by their energy values in the corresponding Summation-FSP-FR energy maps. It should be noted that the deconvolutional operation in a CNN model in back-propagation process will cause offset in the energy map, which leads to Halo Effect around object edges. Thus, we regularize the Summation-FSP-FR energy into the super-pixels generated by the method proposed in [44] to preserve objects' edge.

Algorithm 4 Experimental procedure of weakly supervised semantic segmentation

1: Given a test image and a predicted label by the trained VggNet;
2: Run FSP according to the predicted label and obtain Summation Energy Map;
3: Select "conv5_2" layer as the target layer for FR, set preserved neurons as targets for FR algorithm, and run FR simultaneously to get the FSP-FR energy map;
4: Add Summation Energy Map and FSP-FR energy map to obtain Summation-FSP-FR energy map;
5: Get super-pixels by the algorithm proposed in [44], and let the energy value of each super-pixel be equal to the minimum Summation-FSP-FR energy value within that super-pixel;
6: Run the saliency cut to get the segmentation results.

The quantitative results on over-all validation set are listed in Table 2. We compare the performance of our weakly supervised approach with several state-of-the-art approaches with the same experimental setup, i.e. using only images from Pascal VOC2012 and only image-level labels. The results reveal that our approach largely outperforms previous techniques. Particularly, we achieve 10.76% higher mIOU score than the state-of-the-art approaches and update the best records of 16 classes of Pascal VOC2012. Fig. 15 illustrates some successful examples, where we find that

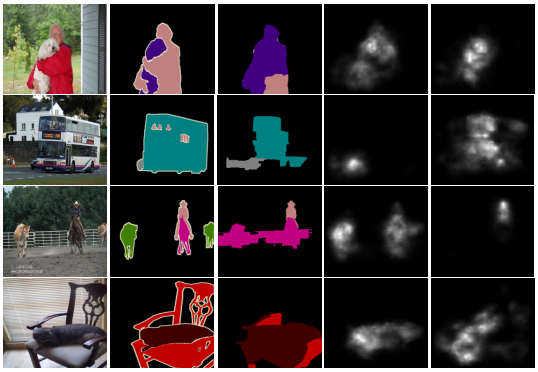


Fig. 16: Failed examples. (a) Input images. (b) Ground truth segmentations. (c) Our segmentation results. (d)(e) Energy maps for different objects generated by our approach.

even for very complex scenes, the proposed approach still works well. Also we show some failure cases and their corresponding objects' energy maps in Fig. 16. We observe that the energy maps are quite meaningful but the segmentation results are not satisfactory. The possible reason derives from the saliency cut [43], which implies that our approach still has the potential to be improved.

5 Conclusion

In this paper, we have proposed the feedback CNN, in which pruning and recovering operations are introduced to implement feedback in deep Convolutional Neural Networks. Feedback CNN achieves the selectivity of neuron activations by jointly reasoning about the outputs of class nodes and the activations of hidden layer neurons. It is capable of capturing high-level semantic concepts

and projects information into image representation as energy maps, based on which recognition, localization and segmentation can be integrated into one unified framework. Qualitative and quantitative experimental results on ImageNet2012 and Pascal VOC2012 have demonstrated the effectiveness of the proposed Feedback CNN.

Due to its importance in both human visual system and machine vision, the feedback mechanism deserves more attention. It still has much space to be further improved, e.g., exploiting neurons in representing multiple objects with the same category, which is critical for instance segmentation. We believe that, with more study of the feedback mechanism, it is highly possible to promote the development in the fields of pattern recognition and computer vision.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[2] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 142–158, 2016.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "Ssd: Single shot multibox detector," *arXiv preprint arXiv:1512.02325*, 2015.

[7] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[8] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1529–1537.

[9] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*. Springer, 2014, pp. 818–833.

[10] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Object detectors emerge in deep scene cnns," *arXiv preprint arXiv:1412.6856*, 2014.

[11] R. Desimone and J. Duncan, "Neural mechanisms of selective visual attention," *Annual Review of Neuroscience*, vol. 18, no. 1, pp. 193–222, 1995.

[12] N. Kruger, P. Janssen, S. Kalkan, M. Lappe, A. Leonardis, J. Piater, A. J. Rodriguez-Sanchez, and L. Wiskott, "Deep hierarchies in the primate visual cortex: What can we learn for computer vision?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1847–1871, 2013.

[13] R. J. Douglas, C. Koch, M. Mahowald, K. A. Martin, and H. H. Suarez, "Recurrent excitation in neocortical circuits," *Science*, vol. 269, no. 5226, p. 981, 1995.

[14] C. Cao, X. Liu, Y. Yang, Y. Yu, J. Wang, Z. Wang, Y. Huang, L. Wang, C. Huang, W. Xu *et al.*, "Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2956–2964.

[15] R. Salakhutdinov and G. E. Hinton, "Deep boltzmann machines." in *International Conference on Artificial Intelligence and Statistics*, vol. 1, 2009, p. 3.

[16] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[17] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034.

[19] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[20] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

[21] C. D. Gilbert and W. Li, "Top-down influences on visual processing," *Nature Reviews Neuroscience*, vol. 14, no. 5, pp. 350–363, 2013.

[22] C. D. Gilbert and M. Sigman, "Brain states: top-down influences in sensory processing," *Neuron*, vol. 54, no. 5, pp. 677–696, 2007.

[23] M. Bar, K. S. Kassam, A. S. Ghuman, J. Boshyan, A. M. Schmid, A. M. Dale, M. S. Hämäläinen, K. Marinkovic, D. L. Schacter, B. R. Rosen *et al.*, "Top-down facilitation of visual recognition," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 2, pp. 449–454, 2006.

[24] P. Hu and D. Ramanan, "Bottom-up and top-down reasoning with convolutional latent-variable models," *arXiv preprint arXiv:1507.05699*, 2015.

[25] Q. Wang, J. Zhang, S. Song, and Z. Zhang, "Attentional neural network: Feature selection using cognitive feedback," in

Advances in Neural Information Processing Systems, 2014, pp. 2033–2041.

[26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[27] M. F. Stollenga, J. Masci, F. Gomez, and J. Schmidhuber, "Deep networks with internal selective attention through feedback connections," in *Advances in Neural Information Processing Systems*, 2014, pp. 3545–3553.

[28] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," in *Advances in Neural Information Processing Systems*, 2014, pp. 2204–2212.

[29] K. Sohn, G. Zhou, C. Lee, and H. Lee, "Learning and selecting features jointly with point-wise gated boltzmann machines." in *ICML (2)*, 2013, pp. 217–225.

[30] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.

[31] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Is object localization for free?-weakly-supervised learning with convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 685–694.

[32] A. Bergamo, L. Bazzani, D. Anguelov, and L. Torresani, "Self-taught object localization with deep networks," *arXiv preprint arXiv:1409.3964*, 2014.

[33] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," *arXiv preprint arXiv:1512.04150*, 2015.

[34] J. Zhang, Z. Lin, J. Brandt, X. Shen, and S. Sclaroff, "Top-down neural attention by excitation backprop," in *European Conference on Computer Vision*. Springer, 2016, pp. 543–559.

[35] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, "What's the point: Semantic segmentation with point supervision," *arXiv preprint arXiv:1506.02106*, 2015.

[36] H.-E. Kim and S. Hwang, "Scale-invariant feature learning using deconvolutional neural networks for weakly-supervised semantic segmentation," *arXiv preprint arXiv:1602.04984*, 2016.

[37] D. Pathak, P. Krahenbuhl, and T. Darrell, "Constrained convolutional neural networks for weakly supervised segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1796–1804.

[38] P. O. Pinheiro and R. Collobert, "From image-level to pixel-level labeling with convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1713–1721.

[39] G. Papandreou, L.-C. Chen, K. P. Murphy, and A. L. Yuille, "Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1742–1750.

[40] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[41] D. M. Beck and S. Kastner, "Top-down and bottom-up mechanisms in biasing competition in the human brain," *Vision Research*, vol. 49, no. 10, pp. 1154–1165, 2009.

[42] R. Desimone, "Visual attention mediated by biased competition in extrastriate visual cortex," *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, vol. 353, no. 1373, pp. 1245–1255, 1998.

[43] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 569–582, 2015.

[44] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 8, pp. 1558–1570, 2015.