Look and Think Twice: Capturing Top-Down Visual Attention with Feedback Convolutional Neural Networks*

Chunshui Cao†  Xianming Liu†  Yi Yang§  Yinan Yu§  Jiang Wang§  Zilei Wang†  Yongzhen Huang‡  Liang Wang‡  Chang Huang‡  Wei Xu‡  Deva Ramanan‡  Thomas S. Huang‡  Baidu Research  Institution of Automation, Chinese Academy of Sciences  Carnegie Mellon University

Abstract

While feedforward deep convolutional neural networks (CNNs) have been a great success in computer vision, it is important to note that the human visual cortex generally contains more feedback than feedforward connections. In this paper, we will briefly introduce the background of feedbacks in the human visual cortex, which motivates us to develop a computational feedback mechanism in deep neural networks. In addition to the feedforward inference in traditional neural networks, a feedback loop is introduced to infer the activation status of hidden layer neurons according to the “goal” of the network, e.g., high-level semantic labels. We analogize this mechanism as “Look and Think Twice.” The feedback networks help better visualize and understand how deep neural networks work, and capture visual attention on expected objects, even in images with cluttered background and multiple objects. Experiments on ImageNet dataset demonstrate its effectiveness in solving tasks such as image classification and object localization.

1. Introduction

“What did you see in this image?”
“Panda, Tiger, Elephant, Lions.”
“Have you seen the Gorilla?”
“Oh! I even didn’t notice there is a Gorilla!”

Visual attention typically is dominated by “goals” from our mind easily in a top-down manner, especially in the case of object detection. Cognitive science explains this in the “Biased Competition Theory” [1, 5, 6], that human visual cortex is enhanced by top-down stimuli and non-relevant neurons will be suppressed in feedback loops when searching for objects. By “looking and thinking twice”, both human recognition and detection performances increase significantly, especially in images with cluttered background [3]. This leads to the selectivity in neuron activations [16], which reduces the chance of recognition being interfered with either noises or distractive patterns.

Inspired by above evidences, we present a novel Feedback Convolutional Neural Network architecture in this paper. It achieves this selectivity by jointly reasoning outputs of class nodes and activations of hidden layer neurons during the feedback loop. As shown in Figure 1 during the feedforward stage, the proposed networks perform inference from input images in a bottom-up manner as traditional Convolutional Networks; while in feedback loops, it sets up high-level semantic labels, (e.g., outputs of class nodes) as the “goal” in visual search to infer the activation status of hidden layer neurons. We show that the network is powerful enough to apply for class model visualization [24, 33], object localization [24] and image classification [15].

Optimization in a Feedback Loop: From a machine learning perspective, the proposed feedback networks add extra flexibility to Convolutional Networks, to help in capturing visual attention and improving feature detection.

*This work is conducted during Chunshui Cao’s and Xianming Liu’s internship at Institute of Deep Learning, Baidu Research.

Figure 1. Feedback Convolutional Net model for capturing visual attention by inferring the status of hidden neuron activations. It is designed to utilize both bottom-up image inputs and top-down semantic labels to infer the hidden neuron activations. Salient areas captured by feedback often correspond to related “target” objects, even in images with cluttered background and multiple objects.
Convolutional Neural Networks [18, 15, 25] have achieved great success in both machine learning and computer vision in recent years. Benefiting from large-scale training data, (e.g., ImageNet [4]), CNNs are capable of learning filters and image compositions at the same time. Various approaches have been adopted to further increase generalization ability of CNNs, by either adding regularisation in training [11, 13], or going deeper [25, 29]. Inspired by Deformable Part-Based Models (DPMs) [8] that characterize middle level part locations as latent variables and search for them during object detection, we utilize a simple yet efficient method to optimize image compositions and assign neuron activations given “goals” in visual search. The algorithm maximizes the posterior response of network given target high-level semantic concepts, in a top-down manner. Compared with traditional bottom-up strategies [11, 13], which aim to regularize the network training, the proposed feedback framework adds flexibilities to the model inference from high-level concepts down to the receptive field.

Figure 2 shows an example on how this flexibility is reflected in detection and localization. Instead of recognizing the input image as a “comic book”, the proposed feedback network is capable to localize each component of the “comic book” via salience maps. The example shown in Figure 1 illustrates its working mechanism: given a high-level semantic stimulus “PANDA”, only the neurons in hidden layers related with the concept “PANDA” will be activated by iterative optimization in a feedback loop. As a result, only salient regions related with the concept “PANDA” are captured in visualizations. As suggested by those results, the feedback networks achieve certain level of selectivity and provide non-relevant suppression during the top-down inference, allowing the model to focus on the most salient image regions that improve the class confidence.

Weakly Supervised Object Localization: Given gradient visualizations in Figure 2, we further develop an algorithm for weakly supervised object localization. Instead of using large amount of supervision (e.g., bounding box positions) in traditional methods such as R-CNN [9] or using regression model [7, 23], we don’t require any localization information in the training stage. Instead, we adopt a unified network performing both recognition and localization tasks, to answer questions of “what” and “where” simultaneously, which are the two most important tasks in computer vision. Experimental results suggest that our weakly supervised algorithm using feedback network could achieve competitive performance on ImageNet object localization task as GoogLeNet [29] and VGG [25].

Image Classification Revisited: We mimic the human visual recognition process that human may focus to recognize objects in a complicated image after a first time glimpse as the procedure “Look and Think Twice” for image classification. We utilize the weakly supervised object localization during the “first glimpse” to make guesses of ROIs, then make the network refocused on those ROIs and give final classifications list. Experimental results on ImageNet 2014 classification validation dataset suggest that this approach is efficient to eliminate irrelevant clutters and improve classification accuracy especially on small objects.

2. Related Work

2.1. Feedforward and Feedback Mechanism

Deep Neural Network takes a feedforward-Back Error Propagation strategy to learn features and classifiers simultaneously, from large scale of training samples [15, 25, 20, 23, 2]. Various approaches have been proposed to further improve the discriminative ability of deep neural network, either by 1) adding regularization to improve the robustness of learnt model and get rid of overfitting, such as Dropout [27], PReLU [11], Batch Normalization [13]; or 2) making the network deeper [29, 25].

Despite great successes achieved by applying Feedforward Networks to image recognition and detection, evidences accumulate from cognitive studies and point to the feedback mechanism that may dominant human perception processes [3, 22, 16, 19]. Recently, tentative efforts have been made to involve feedback strategy into Deep Neural Networks. Deep Boltzmann Machines (DBM) [23, 26] and Deconvolutional Neural Networks [34] try to formulate
the feedback as a reconstruction process within the training stage. Meanwhile, Recurrent Neural Networks (RNN) and Long Short Term Memory (LSTM) [12] are utilized to capture the attention drifting in a dynamic environment and learn the feedback mechanisms via reinforcement learning [25]. DRAW from Google DeepMind [10] combine two into a generative model, to synthesize the image generation process.

As in Biased Competition Theory [6], feedback, which passes the high-level semantic information down to the low-level perception, controls the selectivity of neuron activations in an extra loop in addition to the feedforward process. This results in the “Top-Down” attention in human cognition. Hierarchical probabilistic computational models [19] are proposed to characterize feedback stimuli in a top-down manner, which are further incorporated into deep neural networks, for example, modeling feedback as latent variables in DBM [31], or using selectivity to resolve fine-grained classification [21], et al. However, generative models used in [21, 31] are limited by low computational efficiency and capacity, and are hardly used in large scale datasets.

2.2. Visualization, Detection, and Localization

Feedback is often related with visualization of CNN and object localization since both of them aim to project the high-level semantic information back to image representations. To visualize neuron responses and class models, various approaches are proposed either using deconvolution [29] or optimization based on gradients [24, 17]. As demonstrated in [24], visualization of Convolutional Neural Network shows semantically meaningful salient object regions and helps understand working mechanism of CNNs.

Object detection and localization are more about feedback, by treating detection / localization as a searching process with clear “goals.” To localize and detect objects in images, typical approaches use supervised training, which relies on large amount of supervision, e.g., ground-truth bounding boxes, or manually labeled segmentation in training samples [7]. To behave “searching”, sliding window is used [7], or instead region proposals detected by image segmentations [30] in R-CNN [9]. However, both of these approaches are computational intensive and naturally bottom-up: selecting candidate regions, performing feedforward classification and making decisions.

Inspired by visualizations of CNNs [29, 24], a more feasible and cognitive manner for detection / localization could be derived by utilizing the saliency maps generated in feedback visualizations. Moreover, an ideal approach should unify the recognition and detection in a single feedforward-feedback network architecture. However, if possible, the challenge lies on how to obtain semantically meaningful salience maps with high quality for each concept. That’s the ultimate goal of our work presented in this paper.

3. Model

3.1. Re-interpreting ReLU and Max-Pooling

The most recent state-of-the-art deep CNNs [25] consist of many stacked feedforward layers, including convolutional, rectified linear units (ReLU) and max-pooling layers. For each layer, the input $x$ can be an image or the output of a previous layer, consisting of $C$ input channels of width $M$ and height $N$: $x \in \mathbb{R}^{M \times N \times C}$. The output $y$ consists of a set of $C'$ output channels of width $M'$ and height $N'$: $y \in \mathbb{R}^{M' \times N' \times C'}$.

Convolutional Layer: The convolution layer is used to extract different features of the input. The convolutional layer is parameterized by $C'$ filters with every filter $k \in \mathbb{R}^{K \times K \times C}$.

$$y_{c'} = \sum_{c=1}^{C} k_{c'} \ast x_{c}, \forall c'$$

ReLU Layer: The ReLU layer is used to increase the nonlinear properties of the decision function and of the overall network without affecting the receptive fields of the convolutional layer.

$$y = \max(0, x)$$

Max-Pooling Layer: The max-pooling layer is used to reduce the dimensionality of the output and variance in deformable objects to ensure that the same result will be obtained even when image features have small translations. The max-pooling operation is applied for every pixel $(i,j)$ around its small neighborhood $\mathcal{N}$.

$$y_{i,j,c} = \max_{u,v \in \mathcal{N}} x_{i+u,j+v,c}, \forall i,j,c$$

Selectivity in Feedward Network: To better understand how selectivity works in neural networks and how to formulate the feedback, we re-interpret behaviors of ReLU and Max-Pooling layers as a set of binary activation variables $z \in \{0, 1\}$ instead of the $\max()$ operation in Equations 2 and 3. Thus, behaviors of ReLU and Max-Pooling could be formulated as $y = z \circ x$, where $\circ$ is the element wise product (Hadamard product); and $y = z \ast x$, where $\ast$ is the convolution operator and $z$ is a set of convolutional filters except that they are location variant.

Be interpreting ReLU and Max-Pooling layers as “gates” controlled by input $x$, the network selects information during feedforward phases in a bottom-up manner, and eliminates signals with minor contributions in making decisions. However, the activated neurons could be either helpful or harmful for classification, and involve too many noises, for instance, cluttered backgrounds in complex scenes.

3.2. Introducing the Feedback Layer

Since the model opens all gates and allow maximal information getting through to ensure the generalization,
increase the discriminability within feature level, it is feasible to turn off those gates that provide irrelevant information when targeting at particular semantic labels. This strategy is explained as selectivity in biased competition theory [6] and is critical to realize the top-down attention.

More technically, to increase the model flexibility to images and prior knowledges, we introduce an extra layer to the existing convolutional neural network. We call it the feedback layer. The feedback layer contains another set of binary neuron activation variables $z \in \{0, 1\}$, in addition to ReLU. However, these binary variables are activated by top-down messages from outputs, instead of bottom inputs. The feedback layer is stacked upon each ReLU layer, and they compose a hybrid control unit to active neuron response in both bottom-up and top-down manners:

**Bottom-Up** Inherent the selectivity from ReLU layers, and the dominant features will be passed to upper layers;

**Top-Down** Controlled by Feedback Layers, which propagate the high-level semantics and global information back to image representations. Only those gates related with particular target neurons are activated.

Figure 3 illustrates a simple architecture of our feedback model with only one ReLU layer and one feedback layer.

### 3.3. Updating Hidden Neurons in Feedback Loops

We formulate the feedback mechanism as an optimization problem, by introducing an addition control gate-variable $z$. Given an image $I$ and a neural network with learned parameters $w$, we optimize the target neuron output by jointly inference on binary neuron activations $z$ over all the hidden feedback layers. In particular, if the target neuron is a $k$-th class node in the top layer, we optimize the class score $s_k$ by re-adjusting the neuron activations at every neuron $(i, j)$ of channel $c$, on feedback layer $l$.

$$\max_z s_k(I, z) - \lambda ||z||_1$$

$$s.t. \quad 0 \leq z_{l,i,j,c}^t \leq 1, \forall l, i, j, c$$

(4)

Since the goal of this optimization aims at activating minimal number of neurons yet maximizing the target score, we use $L1$ norm in above target function, as $||z||_1$.

This leads to an integer programming problem, which is NP-hard given the current deep net architecture. An approximation could be derived by applying a linear relaxation:

$$\max_z s_k(I, z) - \lambda ||z||_1$$

$$s.t. \quad 0 \leq z_{l,i,j,c}^t \leq 1, \forall l, i, j, c$$

(5)

We use the gradient ascent algorithm to update the hidden variables through all layers simultaneously.

$$z_{t+1} = z_t + \alpha \cdot \left( \frac{\partial s_k}{\partial z_i} |_{z_t} - \lambda \right)$$

(6)

where $\frac{\partial \lambda ||z||_1}{\partial z_i} = \lambda$ since we require $0 \leq z_{l,i,j,c}^t \leq 1$.

The initialization of feedback layer status $z$ is set to be the corresponding ReLU activation after the first feedforward pass and truncate $z$ when the updated values are either larger than 1 or smaller than 0 during inference.

### 3.4. Implementation Details

As for the implementation details, we set the feedback layer on top of each ReLU layer. We intialize all the hidden activations from $z = 1$, making all feedback “gates” opening during the first time feedforward pass. It is suspected that fully connected layers learn embedding spaces rather than particular parts compared to convolutional layers. We update high level feedback layers according to the sign of gradient of each neuron. We set learning rate of hidden activations to 0.1 and update the neurons of all the feedback layers simultaneously. Each iteration performs a feedforward step of the neural net and a backpropagation step to send back gradients. We stop this process in 10 to 50 iterations.

### 4. Experimental Results

The Feedback Network could be used to improve various computer vision problems. In this paper, we demonstrate...
To conduct qualitative experiments on class neuron visualizations, and quantitative experiments on weakly supervised object localization task. Furthermore, we show that the image recognition could also benefit from the Feedback mechanism, by taking the strategy “Looking and Thinking Twice”, which eliminate noisy or cluttered background and makes the network focused on salient regions. We use three most popular pre-trained ConvNet models, AlexNet [15], VggNet [24] and GoogleNet [29] for experiments. All three models are pre-trained with ImageNet 2012 classification training dataset [4], obtained from Caffe [14] model zoo.

### 4.1. Image Specific Class Model Visualization

Given an image \( I \), a class label \( k \) and the hidden neuron activation states \( z \), we approximate the neural net class score \( s_k \) with the first-order taylor expansion in the neighborhood of \( I \):

\[
s_k(I, z) \approx \mathbf{T}_k(z)^T I + b \tag{7}
\]

where $T_k(z)$ is the derivative of $s_k$ with respect to the image at the point of $I$ and $z$. $T_k(z)$ can be viewed as the linear template applied on image $I$ for measuring how likely the image belongs to class $k$, and could be visualized in the same spatial space since it is of the same size as input image $I$. We use this technique to visualize our feedback model throughout the paper.

More specifically, for a Convolutional Network composed with a stack of piecewise linear layers (i.e., Conv, ReLU and max-pooling) to compute the class scores, once the hidden states $z$ are determined, the final score is a linear function of the image, which is equivalent to the inner product between the template and the image.

Comparison of Visualization Methods: We compare the image gradient (template $T$) after the feedback process against the original one in feedforward pass, and Deconvolutional Neural Networks [33] on a set of complex images containing multiple objects from different classes, with all using the same pre-trained GoogleNet and being given ground truth class labels as a prior. Qualitative results are shown in Figure 4. Without involving the feedback, where all hidden neurons’ status are determined by the bottom-up computation only, the visualization is the same as original image gradient. However, compared with Deconv-like approaches, our feedback model is more efficient in capturing salient regions for each specific class while suppress those irrelevant object areas at the same time after feedback.

Comparison of ConvNet Models: We also qualitatively compare major convolutional network models, i.e., AlexNet, VggNet and GoogleNet, by visualizing their feedback templates in Figure 5. All models are given ground truth labels a prior. From visualizations, we find that VggNet and GoogleNet produce more accurate visual attention than AlexNet, suggesting that using smaller convolution filters and deeper architectures could further distinguish similar and nearby objects. Moreover, although both VggNet and GoogleNet produce very similar image classification accuracies, GoogleNet better captures the salient object areas than VggNet. We hypothesize that the two 4,096 dimensional fully connected layers (i.e., fc6, fc7) in VggNet (which GoogleNet does not contain) could ruin the spatial distinctiveness of image features, as pointed out in [20].

4.2. Weakly Supervised Object Localization

To quantitatively demonstrate the effectiveness of the feedback model, we experiment on the ImageNet 2014 localization task. As pointed in [24], the magnitude of the elements in the model template $T_k$ defines the class-specific salience map on image $I$. Pixels with larger magnitudes indicate that they are more important to the class. We adopt the same saliency extraction strategy as [24] that a single class saliency $M_k$ for class $k$ at pixel $(i,j)$ is computed across all color channels: $M_k(i,j) = \max_{c \in rgb} |T_k(i,j,c)|$.

We show that the proposed Feedback CNN has the potential to unify recognition and detection into a single network architecture in this experiment, instead of using separate ones to perform different tasks respectively. Although the three ConvNets are pre-trained for image classification, we could use the feedbacked salience map for weakly supervised object localization. Given an image and the corresponding class salience map, we compute the object segmentation mask by simply thresholding so that the foreground area covers 95% energy out of the whole salience map, and calculate a tightest bounding box as the localization result. Different from [24], which uses GraphCut [32], this requires saliency maps of higher quality, but only takes less computation.

We test our localization results on ImageNet 2014 validation set, which contains ~ 50,000 images with each image associated with labels and corresponding bounding boxes. A prediction is considered as correct if and only if its overlap with the ground truth bounding box is over 50%. Image are resized to 224x224 to meet the model requirement on resolutions, and ground-truth class labels are provided to predict localizations. Neither further preprocessing nor multi-scale strategy are involved.

Comparison of Localization Methods: Table 1 shows the comparison of our weakly supervised localization accuracy against the baseline method [24]. For fair comparison, we reimplemented the method in [24] following the details

<table>
<thead>
<tr>
<th>Model</th>
<th>Localization Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oxford [24]</td>
<td>44.6</td>
</tr>
<tr>
<td>Feedback</td>
<td>38.8</td>
</tr>
</tbody>
</table>

Table 1. Comparison of our weakly supervised localization results on ImageNet 2014 validation set with the simplified testing protocol: the bounding box is predicted from a single central crop of images and the ground truth labels are provided. We show that our feedback method significant outperforms the baseline method (error rate 44.6%) that uses the original image gradient to localize in [24], both on GoogLeNet architecture.

<table>
<thead>
<tr>
<th>Model</th>
<th>Weakly Supervised Error (%)</th>
<th>Supervised Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlexNet [15]</td>
<td>49.6</td>
<td>-</td>
</tr>
<tr>
<td>VggNet [25]</td>
<td>40.2</td>
<td>34.3 [25]</td>
</tr>
<tr>
<td>GoogleNet [29]</td>
<td><strong>38.8</strong></td>
<td>-</td>
</tr>
</tbody>
</table>

Table 2. Column 2 compares localization errors using feedback on different ConvNet models. VGG and GoogleNet significant outperform AlexNet suggesting they are learning better features. GoogleNet outperforms VGG even further, which matches the observations in Figure 5. We also compare the weakly-supervised feedback mechanism with totally supervised localization model in [25] on VGG, in the third column. It shows that we are competitive to a carefully trained localization model (34.3%) using pixel-wise supervised training data.
in the original paper strictly, and name as “Oxford.” For our method, we use GoogLeNet and apply the same segmentation strategy in our model. Our method obtains 38.8% localization error, and significantly outperforms Oxford (44.6%), suggesting that in terms of capturing attention and localizing salient objects, our feedback net is better. Note that our weakly supervised localization error is even closer to a carefully trained supervised localization model (34.3%).

Comparison of ConvNet Models: We also analyze weakly supervised localization accuracies of above mentioned three ConvNets in Table 2 provided with the same testing protocol. Even provided with ground truth class, VggNet and GoogleNet significantly outperforms AlexNet. This suggests that better feature representations are sharable between the two highly correlated visual tasks: recognition and localization. GoogLeNet outperforms VggNet even further, which matches the observations in Figure 5.

4.3. Image Re-Classification with Attention

Given the weakly supervised attention boxes, the image labels are re-classified using zoomed-in image patches cropped around the bounding boxes. We call such method “Look and Think Twice”, which mimics the human visual recognition process that human may focus to recognize objects in a complicated image after a first time glimpse. We apply this strategy to the image classification. By looking at the full image first in a coarse scale, our model obtains an initial guess of a set of most probable object classes, we then identify the salient object regions from the predicted top-ranked labels using the feedback neural nets, and re-classify those regions. Below are the implementation details on ImageNet 2014 classification:

- Resize image to size 224 * 224, run CNN model and predict top 5 class labels.
- For each of the top 5 class labels, compute object localization box with feedback model.
- Crop image patch for each of 5 bounding boxes from original image and resize to 224 * 224. Predict top 5 labels again.
- Given the total 25 labels and the corresponding confidences, rank them and pick the top 5 as final solution.

Note that when cropping image patches, we use the original image with full resolution (i.e. 375 * 500). This gives the cropped patch enough resolution for recognizing objects in details. Figure 7 shows two exemplar images from ImageNet 2014 validation dataset. Both images are incorrectly classified as “Italian greyhound” and “sunglass” respectively. After feedback, the network determines potential salient ROIs, “reconsider” these salient regions as input and make correct classifications, e.g., “Ibizan hound” and “mask” with high confidence scores. These two examples thoroughly illustrate how looking and thinking twice improves classification tasks.

Classification Accuracy: We test our classification results on ImageNet 2014 validation set, which contains ~
Figure 6. We divide the ImageNet 2014 validation set based on the proportion of the object size in the image. Classification accuracy using feedback crop for images increases with smaller objects. E.g., for those images with object area smaller than 20%, the top-1 classification accuracy increases significantly by almost 5%.

50,000 images with each image associated with one label. Table 3 shows the classification results using a pre-trained GoogleNet on the original full image and on the image patch based on feedback crop. After the re-classification, the top 5 classification errors drops by 1.29%, and, moreover, top 1 error improves even more 1.79%. These results suggest that correct estimations of bounding boxes by a glance can provide more accurate classifications.

Ablative Study: To further understand how “Look and Think Twice” improves the classification task, we divide the ImageNet 2014 validation set based on the proportion of the object size in the image. Figure 6 shows the ablative study. We find that with feedbacks, classification errors drop significantly for images with smaller objects. For example, for objects of less than 20% area of images, the top-1 classification error drops with almost 5%. This suggests traditional ConvNet is powerless in recognizing small objects with cluttered backgrounds, especially when the image is resized to a fixed resolution (i.e. 224 × 224). Our algorithm, on contrary, could focus the network’s attention onto the salient areas, extracting image patches with enough resolutions on the potential objects.

5. Conclusion & Discussion

We propose a Feedback Convolutional Neural Network architecture in this paper, which achieves the top-down selectivity of neuron activations by jointly reasoning the outputs of class nodes and the activations of hidden layer neurons during the feedback loop. The proposed Feedback CNN is capable of capturing high level semantic concepts and project the information down to image representation as salience maps. Benefiting from the feedback mechanism of our model, we utilize the salience map to build a unified deep neural network for both recognition and object localization tasks, to answer questions of both “What” and “Where” simultaneously. Experimental results on ImageNet 2014 object localization Challenge show that our model could achieve competitive performance compared with state-of-the-arts, using only weakly supervised information. We also show the feedback could improve image classification task by re-focusing the network onto those salient regions.

We foresee the potential of Feedback CNN to further improve various computer vision and machine learning tasks, such as fine-grained recognition, object detection, and multi-tasks learning. However, instead of simulating the human vision system, we attribute the improvement of Feedback CNN to the efficiency in utilizing computation resources.

Acknowledgement

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPUs used in the prototyping stage of this research. This research is supported by National Science Foundation under Grant No. 1318971. This work is jointly supported by National Natural Science Foundation of China (61420106015, 61175003) and National Basic Research Program of China (2012CB316300).
References


