International Journal of Computer Vision
https://doi.org/10.1007/s11263-017-1059-x

Top-Down Neural Attention by Excitation Backprop

Jianming Zhang[1] · Sarah Adel Bargal[2] · Zhe Lin[1] · Jonathan Brandt[1] · Xiaohui Shen[1] · Stan Sclaroff[2]

Received: 25 April 2017 / Accepted: 30 November 2017

Abstract

We aim to model the top-down attention of a convolutional neural network (CNN) classifier for generating task-specific attention maps. Inspired by a top-down human visual attention model, we propose a new backpropagation scheme, called Excitation Backprop, to pass along top-down signals downwards in the network hierarchy via a probabilistic Winner-Take-All process. Furthermore, we introduce the concept of contrastive attention to make the top-down attention maps more discriminative. We show a theoretic connection between the proposed contrastive attention formulation and the Class Activation Map computation. Efficient implementation of Excitation Backprop for common neural network layers is also presented. In experiments, we visualize the evidence of a model's classification decision by computing the proposed top-down attention maps. For quantitative evaluation, we report the accuracy of our method in weakly supervised localization tasks on the MS COCO, PASCAL VOC07 and ImageNet datasets. The usefulness of our method is further validated in the text-to-region association task. On the Flickr30k Entities dataset, we achieve promising performance in phrase localization by leveraging the top-down attention of a CNN model that has been trained on weakly labeled web images. Finally, we demonstrate applications of our method in model interpretation and data annotation assistance for facial expression analysis and medical imaging tasks.

Keywords Convolutional neural network · Top-down attention · Selective tuning

1 Introduction

Top-down task-driven attention is an important mechanism for efficient visual search (Usher and Niebur 1996; Wolfe et al. 2003). Various top-down attention models have been proposed, e.g. Koch and Ullman (1987), Anderson and Essen (1987), Tsotsos et al. (1995) and Wolfe (1994). Among them, the Selective Tuning attention model (Tsotsos et al. 1995) provides a biologically plausible formulation. Assuming a pyramidal neural network for visual processing, the Selective Tuning model is composed of a bottom-up sweep of the network to process input stimuli, and a top-down Winner-Take-ALL (WTA) process to localize the most relevant neurons in the network for a given top-down signal. During the top-down process, the Selective Tuning model processes a network as a group of trees (or cones) where every output neuron is the root of a tree (or cone). Selective Tuning localizes the most relevant neurons in the processing cone and generates a binary attention map. Figure 2a illustrates this idea.

Inspired by the Selective Tuning model, we propose a top-down attention formulation for modern CNN classifiers. Instead of the deterministic WTA process used by Tsotsos et al. (1995), which can only generate binary attention maps,

Communicated by Jiri Matas, Bastian Leibe, Max Welling and Nicu Sebe.

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s11263-017-1059-x) contains supplementary material, which is available to authorized users.

✉ Jianming Zhang
jianmzha@adobe.com

Sarah Adel Bargal
sbargal@bu.edu

Zhe Lin
zlin@adobe.com

Jonathan Brandt
jbrandt@adobe.com

Xiaohui Shen
xshen@adobe.com

Stan Sclaroff
sclaroff@bu.edu

[1] Adobe Research, San Jose, CA, USA

[2] Computer Science Department, Boston University, Boston, MA, USA

Published online: 23 December 2017

we formulate the top-down attention of a CNN classifier as a *probabilistic* WTA process.

The probabilistic WTA formulation is realized by a novel backpropagation scheme, called *Excitation Backprop*, which integrates both top-down and bottom-up information to compute the winning probability of each neuron efficiently. Interpretable attention maps can be generated by Excitation Backprop at intermediate convolutional layers, thus avoiding the need to perform a complete backward sweep.

We further introduce the concept of contrastive top-down attention, which captures the differential effect between a pair of contrastive top-down signals. The contrastive top-down attention can significantly improve the discriminativeness of the generated attention maps. We show that contrastive top-down attention can also be computed in less than a complete backward sweep. We further provide a theoretic connection between our contrastive attention formulation and the Class Activation Map computation (Zhou et al. 2016).

The attention maps generated by our probabilistic WTA formulation allow us to visualize the evidence a recognition model uses to make a specific classification decision, whether the decision is a correct or incorrect one. This is important for model analysis; understanding why a network made the correct decision or why it was confused and made a mistake. To quantitatively evaluate these evidence maps, we report their accuracy for the localization task.

In experiments, our method achieves superior weakly supervised localization performance versus Simonyan and Vedaldi (2014), Zeiler et al. (2014), Cao et al. (2015), Zhou et al. (2016) and Bach et al. (2015) on challenging datasets such as PASCAL VOC (Everingham et al. 2010) and MS COCO (Lin et al. 2014). We further explore the scalability of our method for localizing a large number of visual concepts. For this purpose, we train a CNN tag classifier to predict ~18K tags using 6M weakly labeled web images. By leveraging our top-down attention model, our image tag classifier can be used to localize a variety of visual concepts. Moreover, our method can also help to understand what has been learned by our tag classifier. Some examples are shown in Fig. 1.

The performance of our large-scale tag localization method is evaluated on the challenging Flickr30k Entities dataset (Plummer et al. 2015). Without using a language model or any localization supervision, our top-down attention based approach achieves competitive phrase-to-region performance versus a fully-supervised baseline (Plummer et al. 2015).

Finally, we demonstrate promising applications of our method for interpreting neural network models and assisting human data annotators. We discuss several usage examples in facial emotion analysis and medical image analysis.

To summarize, the main contributions of this paper are:

- a top-down attention model for CNN based on a probabilistic Winner-Take-All process using a novel Excitation Backprop scheme;
- a contrastive top-down attention formulation for enhancing the discriminativeness of attention maps;
- a theoretic connection between the contrastive top-down attention formulation and the Class Activation Map computation; and
- a large-scale empirical exploration of weakly supervised text-to-region association by leveraging the top-down neural attention model.

2 Related Work

There is a rich literature about modeling the top-down influences on selective attention in the human visual system (see Baluch and Itti (2011) for a review). It is hypothesized that top-down factors like knowledge, expectations and behav-

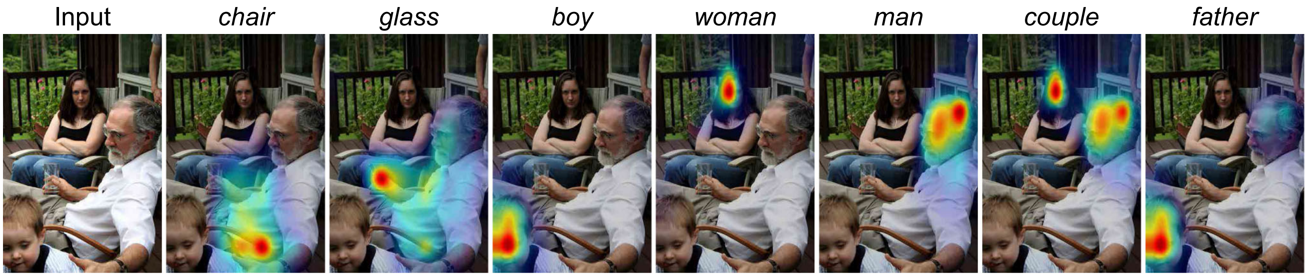


Fig. 1 A CNN classifier's top-down attention maps generated by our Excitation Backprop can localize common object categories, e.g. `chair` and `glass`, as well as fine-grained categories like `boy`, `man` and `woman` in this example image. Our method is capable of performing such localization given a model that was originally trained for recognition and not localization. The classifier used in this example is trained to predict ~18K tags using only weakly labeled web images. Visualizing the classifier's top-down attention can also help interpret what has been learned by the classifier. For `couple`, we can tell that our classifier uses the two adults in the image as the evidence, while for `father`, it mostly concentrates on the child. This indicates that the classifier's understanding of `father` may strongly relate to the presence of a child. This example image is resized to 224 × 224 for our method. The attention maps are generated at a low resolution and upsampled by bicubic interpolation

ioral goals can affect the feature and location expectancy in visual processing (Wolfe 1994; Anne 1980; Koch and Ullman 1987; Desimone and Duncan 1995), and bias the competition among the neurons (Reynolds and Heeger 2009; Tsotsos et al. 1995; Desimone and Duncan 1995; Desimone 1998; Diane 2009). Our attention model is related to the Selective Tuning model of Tsotsos et al. (1995), which proposes a biologically inspired attention model using a top-down WTA inference process.

Various methods have been proposed for grounding a CNN classifier's prediction. Zeiler et al. (2014) and Zhou et al. (2015) use masking-based methods to predict salient image regions. This method slides a mask over the receptive field and uses the score/response decrease as the indicator of the importance of the masked area. Recently, Fong and Vedaldi (2017) use a meta-learning paradigm to predict the minimally salient region by editing the image and learning from the corresponding changes to its output. In Simonyan and Vedaldi (2014), Zeiler et al. (2014) and Springenberg et al. (2014), error backpropagation based methods are used for visualizing relevant regions for a predicted class or the activation of a hidden neuron. Recently, a layer-wise relevance backpropagation method is proposed by Bach et al. (2015) to provide a pixel-level explanation of CNNs' classification decisions. Cao et al. (2015) propose a feedback CNN architecture for capturing the top-down attention mechanism that can successfully identify task-relevant regions. The architecture requires the addition of a binary neuron feedback layer after every ReLU layer. Neurons in the feedback layer pass dominant features to upper layers and propagate high level semantics to lower layers to create attention maps. In Zhou et al. (2016), it is shown that replacing fully-connected layers with an average pooling layer can help generate coarse class activation maps that highlight task relevant regions.

Unlike these previous methods, our top-down attention model is based on the WTA principle, and has an interpretable probabilistic formulation. Our method is also conceptually simpler than Cao et al. (2015) and Zhou et al. (2015) as we do not require modifying a network's architecture or performing additional training. The ultimate goal of our method goes beyond visualization and explanation of a classifier's decision (Zeiler et al. 2014; Springenberg et al. 2014; Bach et al. 2015), as we aim to maneuver CNNs' top-down attention to generate highly discriminative attention maps for the benefits of localization.

Training CNN models for weakly supervised localization has been studied by Oquab et al. (2015), Pathak et al. (2015), Papandreou et al. (2015), Pinheiro and Collobert (2015), Fang et al. (2015), Simonyan and Vedaldi (2014), Guillaumin et al. (2014) and Bazzani et al. (2016). In Oquab et al. (2015), Fang et al. (2015) and Pinheiro and Collobert (2015), a CNN model is transformed into a fully convolutional net to perform efficient sliding window inference, and then Multiple Instance Learning (MIL) is integrated in the training process through various pooling methods over the confidence score map. Due to the large receptive field and stride of the output layer, the resultant score maps only provide very coarse location information. To overcome this issue, a variety of strategies, e.g. image re-scaling and shifting, have been proposed to increase the granularity of the score maps (Oquab et al. 2015; Pinheiro and Collobert 2014, 2015). Image and object priors are also leveraged to improve the object localization accuracy in Pathak et al. (2015), Papandreou et al. (2015) and Pinheiro and Collobert (2015). Guillaumin et al. (2014) perform weakly supervised localization using appearance models of previously localized (segmented) classes to select and segment a new class, thereby deriving a binary segmentation mask for each image. Compared with weakly supervised localization, the problem setting of our task is essentially different. We assume a pre-trained deep CNN model is given, which may not use any dedicated training process or model architecture for the purpose of localization. Our focus, instead, is to model the top-down attention mechanism of *generic* CNN models to produce interpretable and useful task-relevant attention maps.

3 Methods

3.1 Top-Down Neural Attention Based on Probabilistic WTA

We consider a generic feedforward neural network model. The goal of a top-down attention model is to identify the task-relevant neurons in the network.

Given a selected output unit, a deterministic top-down WTA scheme is used in the biologically inspired Selective Tuning model (Tsotsos et al. 1995) to localize the most relevant neurons in the processing cone (see Fig. 2a) and generate a binary attention map. Inspired by the deterministic WTA, we propose a *probabilistic* WTA formulation to model a neural network's top-down attention (Fig. 2b, c), which leverages more information in the network and generates soft attention maps that can capture subtle differences between top-down signals. This is critical to our contrastive attention formulation in Sect. 3.3.

In our formulation, the top-down signal is specified by a prior distribution $P(A_0)$ over the output units, which can model the uncertainty in the top-down control process. Then the winner neurons are recursively sampled in a top-down fashion based on a conditional winning probability $P(A_t|A_{t-1})$, where $A_t, A_{t-1} \in \mathcal{N}$ denote the selected winner neuron at the current and the previous step respectively, and $\mathcal{N}$ is the overall neuron set. We formulate the top-down relevance of each neuron as its probability of being selected

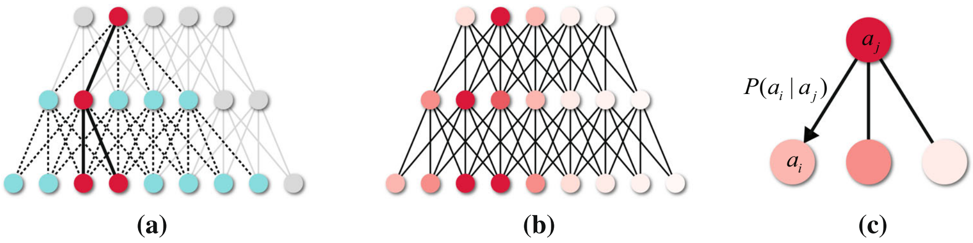


Fig. 2 Deterministic WTA (Tsotsos et al. 1995) versus our probabilistic WTA for modeling top-down attention. **a** Given a selected output unit, the red dots denote the winners identified by the top-down layer-wise deterministic WTA scheme in the processing cone, and the cyan ones are inhibited. **b** In our probabilistic WTA scheme, winner neurons are generated by a stochastic sampling process (shown in (**c**)). The top-down signal is specified by a probability distribution over the output units. The shading of a dot in (**b**) indicates the its relative likelihood of winning against the other ones in the same layer. This figure contrasts the deterministic Selective Tuning Model and our probabilistic model. **a** Deterministic WTA, **b** probabilistic WTA and **c** winner sampling

as a winner in this process. Formally, given a neuron $a_j \in \mathcal{N}$ (note that a_j denotes a specific neuron and A_t denotes a variable over the neurons), we would like to compute its *Marginal Winning Probability* (MWP) $P(a_j)$. The MWP $P(a_j)$ can be factorized as

$$P(a_j) = \sum_{a_i \in \mathcal{P}_j} P(a_j|a_i)P(a_i), \tag{1}$$

where $\mathcal{P}_j$ is the parent node set of a_j (in top-down order). As Eq. 1 indicates, given $P(a_j|a_i)$, $P(a_j)$ is a function of the marginal winning probability of the parent nodes in the preceding layers. It follows that $P(a_j)$ can be computed in a top-down layer-wise fashion.

Our formulation is equivalent to an absorbing Markov chain process (Kemeny et al. 1960). A Markov Chain is an absorbing chain if (1) there is at least one absorbing state and (2) it is possible to go from any state to at least one absorbing state in a finite number of steps. Any walk will eventually end at one of the absorbing states. Non-absorbing states are called Transient States. In our formulation, each random walk starts from an output neuron and ends at some absorbing node of the bottom layer in the network.

For an absorbing Markov Chain, the canonical form of the transition matrix P can be represented by

$$P = \begin{bmatrix} Q & R \\ \mathbf{0} & I_r \end{bmatrix}, \tag{2}$$

where the entry p_{ij} is the transition probability from state i to j. Each row sums up to one and I_r is an $r \times r$ identity matrix corresponding to the r absorbing states. Please refer to the supplementary material for more details on the form of matrices Q and R. In our formulation, $p_{ij} := P(a_j|a_i)$ is the transition probability.

The fundamental matrix of the absorbing Markov chain process is

$$N = \sum_{k=0}^{\infty} Q^k = (I_t - Q)^{-1}. \tag{3}$$

The (i, j) entry of N can be interpreted as the expected number of visits to node j, given that the walker starts at i. In our formulation, the MWP $P(a_j)$ can then be interpreted as the expected number of visits when a walker starts from a random node of the output layer according to $P(A_0)$. This expected number of visits can be computed by a simple matrix multiplication using the fundamental matrix of the absorbing Markov chain. In this light, the MWP $P(a_i)$ is a linear function of the top-down signal $P(A_0)$, which will be shown to be convenient for computing linear combinations of multiple MWP maps (see Sect. 3.3). In practice, our Excitation Backprop does the computation in a layer-wise fashion, without the need to explicitly construct the fundamental matrix. This layer-wise propagation is possible due to the acyclic nature of the feedforward network.

3.2 Excitation Backprop

In this section, we propose the Excitation Backprop method to realize the probabilistic WTA formulation for modern CNN models.

A modern CNN model (Krizhevsky et al. 2012; Simonyan and Zisserman 2015; Szegedy et al. 2015) is mostly composed of a basic type of neuron a_i, whose response is computed by $\widehat{a}_i = \varphi(\sum_j w_{ji}\widehat{a}_j + b_j)$. Here w_{ji} is the weight, $\widehat{a}_j$ is the input, b_j is the bias and φ is the nonlinear activation function. We call this type of neuron an *Activation Neuron*. We have the following assumptions about the activation neurons.

A1 The response of the activation neuron is non-negative.
A2 An activation neuron is tuned to detect certain visual features. Its response is positively correlated to its confidence of the detection.

A1 holds for a majority of the modern CNN models, as they adopt the Rectified Linear Unit (ReLU) as the activation function. **A2** has been empirically verified by many recent works (Zhou et al. 2015; Zeiler et al. 2014; Zhou et al.

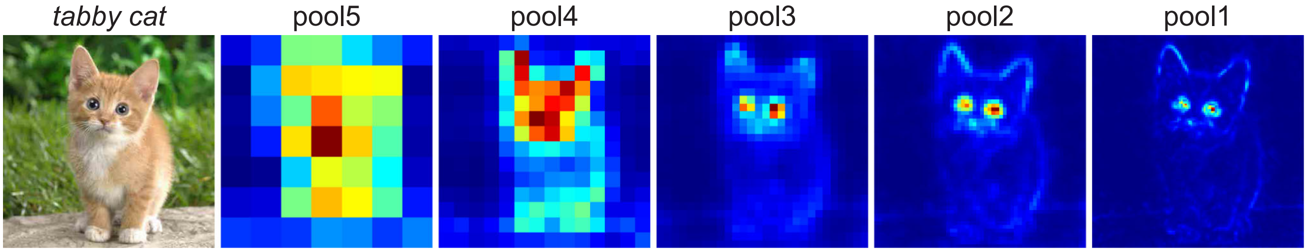


Fig. 3 Example Marginal Winning Probability (MWP) maps computed via Excitation Backprop from different layers of the public VGG16 model (Simonyan and Zisserman 2015) trained on ImageNet. The input image is shown on the left. The MWP maps are generated for the category `tabby cat`. Neurons at higher-level layers have larger receptive fields and strides. Thus, they can capture larger areas but with lower spatial accuracy. Neurons at lower layers tend to more precisely localize features at smaller scale. An H × W × D attention map is converted to a 224 × 224 × 1 image for visualizing the MWP by first averaging the channels, followed by performing nearest neighbor interpolation

2014; Yosinski et al. 2015; Gonzalez-Garcia et al. 2016). It is observed that neurons at lower layers detect simple features like edge and color, while neurons at higher layers can detect complex features like objects and body parts.

Given **A1** and **A2**, it follows that negative weights in the network will always decrease the corresponding activation values, because

$$\widehat{a}_i = \varphi\left(\sum_{j,w_{ji}\geq 0} w_{ji}\widehat{a}_j + \sum_{j,w_{ji}<0} w_{ji}\widehat{a}_j + b_j\right) \tag{4}$$

$$\leq \varphi\left(\sum_{j,w_{ji}\geq 0} w_{ji}\widehat{a}_j + b_j\right) \quad \text{if } \widehat{a}_j \geq 0 \tag{5}$$

Therefore, between activation neurons, we define a connection to be *excitatory* if its weight is non-negative, and *inhibitory* otherwise. Our Excitation Backprop passes top-down signals through excitatory connections between activation neurons. Formally, let $\mathcal{C}_i$ denote the child node set of a_i (in the top-down order). For each $a_j \in \mathcal{C}_i$, the conditional winning probability $P(a_j|a_i)$ is defined as

$$P(a_j|a_i) = \begin{cases} Z_i\widehat{a}_j w_{ji} & \text{if } w_{ji} \geq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

$Z_i = 1/\sum_{j:w_{ji}\geq 0}\widehat{a}_j w_{ji}$ is a normalization factor so that $\sum_{a_j\in\mathcal{C}_i} P(a_j|a_i) = 1$. In the special case when $\sum_{j:w_{ji}\geq 0}\widehat{a}_j w_{ji} = 0$, we define Z_j to be 0. Note that the formulation of $P(a_j|a_i)$ is valid due to **A1**, since $\widehat{a}_j$ is always non-negative.

Equation 6 assumes that if a_i is a winner neuron, the next winner neuron will be sampled among its child node set $\mathcal{C}_i$ based on the connection weight w_{ji} and the input neuron's response $\widehat{a}_j$. The weight w_{ji} captures the top-down feature expectancy, while $\widehat{a}_j$ represents the bottom-up feature strength, as assumed in **A2**. Due to **A1**, child neurons of a_i with negative connection weights always have an inhibitory effect on a_i, and thus are excluded from the competition.

Equation 6 recursively propagates the top-down signal layer by layer, and we can compute attention maps from any intermediate convolutional layer. For our method, we simply take the sum across channels to generate a marginal winning probability (MWP) map as our attention map, which is a 2D probability histogram. Figure 3 shows some example MWP maps generated using the pre-trained VGG16 model (Simonyan and Zisserman 2015). Neurons at higher-level layers have larger receptive fields and strides. Thus, they can capture larger areas but with lower spatial accuracy. Neurons at lower layers tend to more precisely localize features at smaller scales. Note that doing Excitation Backprop at the pixel level does not have interpretable meanings, since the pixel values cannot be treated as activation values defined in **A2**. Moreover, the mean-subtracted pixel values can be negative, violating **A1**.

3.3 Contrastive Top-Down Attention

Since the MWP is a linear function of the top-down signal (see Sect. 3.1), we can compute any linear combination of MWP maps for an image by a single backward pass. All we need to do is linearly combine the top-down signal vectors at the top layer before performing a *single* Excitation Backprop pass. In this section, we take advantage of this property to generate highly discriminative top-down attention maps by passing down one signal that is equivalent to a pair of contrastive signals.

For each output unit o_i, we construct a reverse unit $\bar{o}_i$, whose input weights are the negation of those of o_i. For example, if an output unit corresponds to an `elephant` classifier, then its reverse unit will correspond to a `non-elephant` classifier. Subtracting the MWP map for `non-elephant` from the one for `elephant` will cancel out common winner neurons and amplify the discriminative neurons for `elephant`. The resulting map is able to highlight the unique

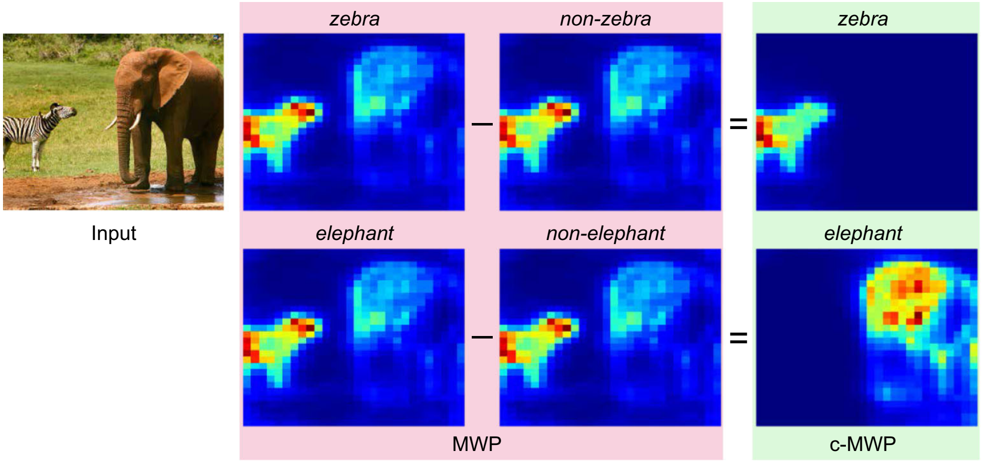


Fig. 4 Marginal Winning Probability (MWP) versus contrastive MWP (c-MWP). The input image is resized to 224 × 224, and we use VGG16 pretrained on ImageNet to generate the MWP maps for `zebra` and `elephant`, as well as `non-zebra` and `non-elephant`. The c-MWP maps are computed by subtracting the `non-zebra` (`non-elephant`) MWP map from the `zebra` (`elephant`) MWP map, and then thresholding the values at 0. All attention maps are rescaled for visualization. It is surprising that all the MWP maps shown above look nearly identical. Their subtle differences are captured by the c-MWP maps, where the common winner neurons for different concepts are cancelled out

features of `elephant`. We call this map a *contrastive* MWP (c-MWP) map, which can be computed by a single backward pass. Figure 4 shows some examples.

Formally, let W_1 be the weights of the top layer, and P_1 be the corresponding transition matrix whose entries are the conditional probabilities defined by Eq. 6. Suppose the number of the neurons at the top is m and at the next lower layer is k. Then P_1 is a $m \times k$ matrix, and the input top-down signal P_0 is a k-D vector with k equal to the number of classes for which the network was trained. The MWP of a target layer, say the n-th layer from the top, is formulated as

$$C = P_{n-1} \cdot \ldots \cdot P_2 \cdot P_1 \cdot P_0, \tag{7}$$

For the contrastive output units, the weights are the negation of the original weights at the top layer, namely $-W_1$. Since the top layer is a linear classification layer, negating its weights is equivalent to flipping the normal vector of the classification hyperplane. Let $\bar{P}_1$ denote the transition matrix of the negated classification layer. Regarding $\bar{P}_1$, the entries that are positive were previously thresholded in P_1 according to Eq. 6 and *vise versa*. For example, $p_{ij} > 0$ in P_1 indicates $\bar{p}_{ij} = 0$ in $\bar{P}_1$. Then the MWP of the contrastive output units is

$$\bar{C} = P_{n-1} \cdot \ldots \cdot P_2 \cdot \bar{P}_1 \cdot P_0. \tag{8}$$

The resultant contrastive MWP is formulated as

$$C - \bar{C} = P_{n-1} \cdot \ldots \cdot P_2 \cdot (P_1 - \bar{P}_1) \cdot P_0. \tag{9}$$

In practice, we compute $P_0 \cdot P_1$ and $P_0 \cdot \bar{P}_1$ respectively by Excitation Backprop. Then, we do the subtraction and propagate the contrastive signals $P_0 \cdot (P_1 - \bar{P}_1)$ downwards by Excitation Backprop again. Moreover, we truncate the contrastive MWP map at zero so that only positive parts are kept. Our probabilistic formulation ensures that there are always some positive parts on the contrastive MWP map, unless the MWP map and its reverse map are identical.

4 Some Discussion

In this section, we first compare CAM (Zhou et al. 2016) and contrastive MWP. We then explore some drawbacks of previous gradient-based methods for visualizing top-down saliency.

4.1 CAM and Contrastive MWP

Many recent CNN classifiers use the Global Average Pooling (GAP) layer to generate a global feature vector for the whole image. Then, a linear classification layer, namely a fully-connected layer, is placed right after the GAP layer. For this particular architecture design, Zhou et al. (2016) proposed the Class Activation Map (CAM) method to generate coarse top-down saliency maps. The basic idea of the CAM method is to remove the GAP layer and treat the model as a fully-convolutional architecture (Long et al. 2015). The last classification layer is then applied on the feature map right before the GAP layer as a 1 × 1 filter. The resulting

Alg. 1 Excitation Backprop for the Convolutional/Fully-connected Layer

```
input   : $A_n$: bottom activation responses;
          $W$: weight parameters;
          $P_{n-1}$: top MWP
output  : $P_n$: bottom MWP
compute $W^+$ by thresholding $W$ at zero
compute $X = W^{+T} A_n$ in Eq. 18 (a forward layer operation in Caffe)
compute $Y = P_{n-1} \oslash X$ by element-wise division
compute $Z = W^+ Y$ (a backward layer operation in Caffe)
compute $P_n = A_n \odot Z$ by element-wise multiplication
```

dense predictions are the class activation maps, which can provide discriminative localization for related categories.

In this section, we show a theoretic connection between the Class Activation Maps and the proposed contrastive MWP maps when the CNN classifier uses the GAP layer right before the classification layer.

For a target category, let (W, b) denote the weights and bias of the corresponding linear classifier of the last classification layer. $W = [w_i]_n$ is a n-d vector and b is a scalar value. The output of the GAP layer is also a n-d vector $[a_i]_n$, where

$$a_i = \frac{1}{d}\sum_j a_{ij}. \tag{10}$$

$[a_{ij}]_{n\times d}$ denotes the n-channel feature map that is input to the GAP layer, and $j = 1 \ldots d$ represents the spatial index.

Let us first take a look at how to compute the contrastive MWP map regarding $[a_{ij}]_{n\times d}$. Based on our contrastive MWP formulation, the contrastive MWP $[c_i]_n$ regarding the GAP layer is computed by

$$c_i = \frac{w_i^+ a_i}{\sum_i w_i^+ a_i} - \frac{w_i^- a_i}{\sum_i w_i^- a_i}. \tag{11}$$

Here $w_i^+ = \max\{w_i, 0\}$ and $w_i^- = \max\{-w_i, 0\}$, and we assume that the initial top-down signal distribution for the target category is 1. After the contrastive top-down signals pass though the GAP layer, we will get the contrastive MWP $[c_{ij}]_{n\times d}$ for $[a_{ij}]_{n\times d}$. Based on Eqs. 10 and 11,

$$\begin{aligned} c_{ij} &= \frac{a_{ij}}{\sum_j a_{ij}} c_i, \\ &= \frac{a_{ij}}{d \cdot a_i}\left(\frac{w_i^+ a_i}{\sum_i w_i^+ a_i} - \frac{w_i^- a_i}{\sum_i w_i^- a_i}\right), \\ &= \frac{1}{d}\left(\frac{w_i^+ a_{ij}}{\sum_i w_i^+ a_i} - \frac{w_i^- a_{ij}}{\sum_i w_i^- a_i}\right). \end{aligned} \tag{12}$$

As described before, to get a 2-d map $[C_j]_d$, we will just sum the signals along the channel. We have

$$\begin{aligned} C_j &= \sum_i c_{ij}, \\ &= \frac{1}{d}\left(\frac{w_i^+ \sum_i a_{ij}}{\sum_i w_i^+ a_i} - \frac{w_i^- \sum_i a_{ij}}{\sum_i w_i^- a_i}\right), \\ &= Z^+ w_i^+ \sum_i a_{ij} - Z^- w_i^- \sum_i a_{ij}, \end{aligned} \tag{13}$$

where

$$Z^+ = \frac{1}{d\sum_i w_i^+ a_i}, \tag{14}$$

$$Z^- = \frac{1}{d\sum_i w_i^- a_i}. \tag{15}$$

Now let us take a look at CAM. To compute CAM regarding $[a_{ij}]_{n\times d}$, we simply apply the classifier (W, b) densely on the n-channel feature map $[a_{ij}]_{n\times d}$. Let $[S_j]_d$ denote the CAM, and we have

$$\begin{aligned} S_j &= \sum_i w_i a_{ij} + b, \\ &= w_i^+ \sum_i a_{ij} - w_i^- \sum_i a_{ij} + b. \end{aligned} \tag{16}$$

Comparing Eqs. 13 and 16, we can see that the contrastive MWP map equals the CAM when $Z^+ = Z^-$ up to a scale factor (we ignore the constant bias b as it will not affect the ordering the map values). In a sense, Z^+ and Z^- act as normalization terms to balance the positive and the negative top-down signals so that their sum will be the same, namely $Z^+ \sum_i \left(w_i^+ \sum_i a_{ij}\right) = Z^- \sum_j \left(w_i^- \sum_i a_{ij}\right)$.

Therefore, we can expect that for networks using the GAP layer, the performance of the contrastive MWP and the CAM method will be similar if we do Excitation Backprop only to the feature map before the GAP layer. In Sect. 6.1.3, we will verify this hypothesis. This connection between the contrastive MWP and the CAM method also indicate that better localization performance might be achieved by our method when the CNN model uses the GAP layer. This is because the first few steps of the contrastive signal Excitation Backprop will approximate the CAM method, which has a clear dense prediction interpretation.

Compared with the CAM method, our contrastive MWP is more general and it can work with models that do not use the GAP layer. Moreover, we will show that being able to backpropagate the signals further downwards in the network is generally beneficial in terms of localization accuracy (see Table 1).

4.2 What is Wrong with the Gradient-Based Methods?

Several gradient-based methods have been used for visualizing top-down signals (Simonyan and Vedaldi 2014; Zeiler et al. 2014; Springenberg et al. 2014). In this section, however, we argue that the neurons' gradients are generally not correlated with the top-down saliency. In the following, we will list a few examples where gradients give counter-intuitive results for measuring top-down saliency.

4.2.1 Single Linear Layer

In the simplest scenarios where the CNN model has only one layer, i.e. a linear classification model, the gradients of the input feature are completely input-independent. A linear layer is of the form $ax + b$, for which the derivative regarding x will always be a. However, the top-down saliency should be a function of the input feature and the top-down signal.

4.2.2 Average Pooling Layer

An Average Pooling layer is a special case of a linear layer with constant weights. When backpropagating the gradients through an Average Pooling layer, we will not get any spatial information from the gradients. The resulting effect is illustrated in Fig. 6, where we visualize the gradient magnitude maps regarding the layer right before the Average Pooling layer.

4.2.3 Activation Layers

For many common activation layers like tanh and sigmoid, their derivatives are not positively correlated with the activation outputs. Very high activation values may correspond to nearly zero derivatives. When such activation layers are used, the gradient backpropagation will not be able to highlight the regions that result in strong responses of the filters.

In practice, the gradient based methods can still provide some localization information for modern CNN models (Simonyan and Vedaldi 2014; Zeiler et al. 2014; Springenberg et al. 2014). Note that modern CNN models usually use the Rectifier Linear Unit (ReLU) as the activation function and the Max Pooling layer to reduce the feature map dimension. The derivatives of the ReLU layer and the Max Pooling layer are basically binary values indicating the on/off status of the input neurons. Therefore, gradients in this case can contain some spatial information of activated neurons. However, the problems discussed above can limit the robustness and the generalizability of the gradient based methods.

5 Implementation of Excitation Backprop

We implement Excitation Backprop in Caffe (Jia et al. 2014) (We make our code publicly available at our project website[1]). In the following, we describe the implementation of Excitation Backprop for common layers in modern CNNs.

5.1 Convolutional, Fully-Connected and Average Pooling Layers

Convolutional, Fully-connected and Average pooling layers can be regarded as the same type of layers that perform an affine transform of the response values of the bottom (input) neurons. The implementation of Excitation Backprop for these layers exactly follows Eqs. 1 and 6 in our paper. Let $p_{ij} := P(a_j|a_i)$ and $p_j := P(a_j)$ for convenience, and we have:

$$\begin{aligned} p_j &= \sum_{i \in \mathcal{P}_j} p_{ij} p_i \\ &= \sum_{i \in \mathcal{P}_j} Z_i \widehat{a}_j w^+_{ji} p_i \\ &= \widehat{a}_j \sum_{i \in \mathcal{P}_j} w^+_{ji} p_i \frac{1}{\sum_{k \in \mathcal{C}_i} w^+_{ki} \widehat{a}_k}, \end{aligned} \tag{17}$$

where $w^+_{ji} = \max\{w_{ji}, 0\}$, $\mathcal{P}_i$ is the parent node set of a_j and $\mathcal{C}_i$ is the child node set of a_i (in top-down order). The computation of all p_j in a layer can be performed by matrix operations:

$$P_n = A_n \odot \left(W^+ \left(P_{n-1} \oslash (W^{+T} A_n) \right) \right), \tag{18}$$

where P_{n-1} and P_n denote the Marginal Winning Probability (MWP) for the top neurons and the bottom neurons of the layer respectively, and $W^+ = \left[w^+_{ij} \right]_{d_1, d_2}$ is a $d_1 \times d_2$ weight matrix representing the excitatory connection weight of the layer. $d_1 (d_2)$ equals the number of the bottom (top) neurons. A_n is the response value of the bottom neurons. Note that we assume that the response values of the bottom neurons are non-negative; thus, we do not propagate the top-down signals to the mean-subtracted pixel layer, which may contain negative pixel values. Moreover, $\odot$ and $\oslash$ are the element-wise multiplication and division respectively. Algorithm 1 summarizes the steps of Excitation Backprop for the Convolutional/Fully-connected layer. The Average Pooling layer can be regarded as a special case where the weight matrix is composed of constant weights, i.e. $w_{ij} = 1/n$, where n is the number of neurons at each feature channel that are averaged.

[1] http://www.cs.bu.edu/groups/ivc/excitation-backprop.

5.2 Element-wise Sum Layer

The deep Residual Network (ResNet) (He et al. 2016) is a recent architecture for convolutional neural network models. The ResNet features a special residue structure which uses the Element-wise Sum layer to combine a layer with its residual. Let the pair of the input layers' activation values be $[\widehat{a}_i]$ and $[\widehat{r}_i]$, the output of the Element-wise Sum layer is $[\widehat{c}_i] = [\widehat{a}_i + \widehat{r}_i]$ According to Eq. 6, the Excitation Backprop for the Element-wise Sum layer is implemented based on the following formulation:

$$\begin{aligned} P(a_i) &= \frac{\widehat{a}_i}{\widehat{c}_i} P(c_i), \\ P(r_i) &= \frac{\widehat{r}_i}{\widehat{c}_i} P(c_i). \end{aligned} \tag{19}$$

5.3 ReLU Layer

The ReLU layer serves as a neuron-wise gating function. The top-down signals remain the same after a ReLU layer in Excitation Backprop, since each ReLU neuron only has a single child node. Neurons with zero activation values will not be selected as winning neurons due to Eq. 6 in our paper. Thus, the propagation method for ReLU is the same as error backpropagation.

For a class of activation functions that is lower bounded, e.g. the sigmoid function, tanh function and the Exponential Linear Unit (ELU) function (Clevert et al. 2016), we may slightly modify our formulation of Excitation Backprop. Suppose λ is the minimum value the activation function can take. The modified formulation corresponding to Eq. 6 in our paper is

$$P(a_j|a_i) = \begin{cases} Z_i(\widehat{a}_j - \lambda)w_{ji} & \text{if } w_{ji} \geq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

Because $\widehat{a}_j - \lambda \geq 0$, our probability formulation still holds. We postulate that this formulation may work in practice if **A2** still holds. We leave further investigation of this generalized formulation for the future work.

5.4 Max Pooling Layer

Again, as in error backpropagation, signals are copied to the lower layer through the pooling mask, because each pooled neuron has only a single child node. Therefore, the propagation method for the Max Pooling layer is the same as error backpropagation, too.

5.5 Local Response Normalization (LRN) Layer

Special care should be taken for the LRN layer, whose response is computed element-wise as $\widehat{m}_i = s_i \widehat{a}_i$, where s_i is a positively valued scaling factor computed based on the neighboring neurons' response. The LRN layer locally normalizes each neuron's response. In Excitation Backprop, we just ignore the normalization factor, and thus each neuron m_i of the LRN layer has only one child node a_i in the top-down propagation process. As a result, the top-down signals remain the same when passing through the LRN layer.

6 Experiments

6.1 The Pointing Game

The goal of this section is to evaluate the *discriminativeness* of top-down attention maps for localizing target objects in crowded visual scenes.

6.1.1 Experiment Setting

Given a pre-trained CNN classifier, we test different methods in generating a top-down attention map for a target object category present in an image. Ground truth object labels are used to cue the method. We extract the maximum point on the top-down attention map. A hit is counted if the maximum point lies on one of the annotated instances of the cued object category, otherwise a miss is counted. We measure the localization accuracy by $Acc = \frac{\#Hits}{\#Hits + \#Misses}$ for each object category. The overall performance is measured by the mean accuracy across different categories.

We call this the *Pointing Game*, as it asks the CNN model to point at an object of designated category in the image. The pointing game does not require highlighting the full extent of an object, and it does not account for the CNN model's classification accuracy. Therefore, it purely compares the *spatial selectiveness* of the top-down attention maps. Moreover, the pointing game only involves minimum post-processing of the attention maps, so it can evaluate different types of attention maps more fairly.

Datasets We use the test split of the PASCAL VOC07 dataset (Everingham et al. 2010) (4952 images) and the validation split of the MS COCO dataset (Lin et al. 2014) (40,137 images). In particular, COCO contains 80 object categories, and many of its images have multiple object categories, making even the simple Pointing Game rather challenging. To evaluate success in the Pointing Game, we use the ground truth bounding boxes for VOC07 and the provided segmentation masks for COCO.

As the Pointing Game is trivial for images with large dominant objects, we also report the performance on a difficult subset of images for each category. The difficult set includes images that meet two criteria: (1) the total area of bounding boxes (or segments in COCO) of the testing category is smaller than 1/4 the size of the image and (2) there is at least one other distracter category in the image. A link to the difficult sets is available on our github page.[2]

CNN Classifiers We consider four popular CNN architectures: CNN-S (Chatfield et al. 2014) (an improved version of AlexNet (Krizhevsky et al. 2012)), VGG16 (Simonyan and Zisserman 2015), GoogleNet (Szegedy et al. 2015) and ResNet50 (He et al. 2016). These models vary a lot in depth and structure. We download these models from the Caffe Model Zoo website.[3] These models are pre-trained on ImageNet (Russakovsky et al. 2015). For both VOC07 and COCO, we use the training split to fine-tune each model. We follow the basic training procedure for image classification. Only the output layer is fine-tuned using the multi-label cross-entropy loss for simplicity, since the classification accuracy is not our focus. Images are padded to square shape by mirror padding and up-sampled to 256×256. Random flipping and cropping are used for data augmentation. No multi-scale training (Oquab et al. 2015) is used. We fix the learning rate to be 0.01 for all the architectures and optimize the parameters using SGD with momentum=0.9. The training batch size is set as 64, 32, 64 and 32 for VGGS, VGG16, GoogleNet and ResNet50 respectively. We stop the training when the training error plateaus.

Test Methods We compare Excitation Backprop (MWP and c-MWP) with the following methods: (Grad) the error backpropagation method (Simonyan and Vedaldi 2014), (Deconv) the deconvolution method originally designed for internal neuron visualization (Zeiler et al. 2014), (LRP) layer-wise relevance propagation (Bach et al. 2015), and (CAM) the class activation map method (Zhou et al. 2016). We implement Grad, Deconv and CAM in Caffe. For Deconv, we use an improved version proposed in Springenberg et al. (2014), which generates better maps than the original version (Zeiler et al. 2014). For Grad and Deconv, we follow Simonyan and Vedaldi (2014) to use the maximum absolute value across color channels to generate the final attention map. Taking the mean instead of maximum will degrade their performance. For LRP, we use the software provided by the authors, which only supports CPU computation. For VGG16, this software can take 30s to generate an attention map on an Intel Xeon 2.90 GHz × 6 machine.[4] Due to limited computational resources, we do not evaluate LRP for VGG16. Moreover, the implementation of LRP does not support ResNet as it is based on an older version of Caffe. Therefore, we do not evaluate LRP for ResNet50, either.

Note that CAM is only applicable to certain architectures like GoogleNet and ResNet50, which do not have fully connected layers. At test time, it acts like a fully convolutional model to perform dense sliding window evaluation (Oquab et al. 2015; Sermanet et al. 2014). Therefore, the comparison with CAM encompasses the comparison with the dense evaluation approach for weakly supervised localization (Oquab et al. 2015).

To generate the full attention maps for images of arbitrary aspect ratios, we convert each testing CNN classifier to a fully convolutional architecture as in Oquab et al. (2015). All the compared methods can be easily extended to fully convolutional models. In particular, for Excitation Backprop, Grad and Deconv, the output confidence map of the target category is used as the top-down signal to capture the spatial weighting. However, all input images are resized to 224 in the smaller dimension, and no multi-scale processing is used.

For different CNN classifiers, we empirically select different layers to compute our attention maps based on a held-out set. We use the conv5 layer for CNN-S, pool4 for VGG16, pool2 for GoogleNet and res4f for ResNet50. We use bicubic interpolation to upsample the generated attention maps. The effect of the layer selection will be analyzed below. For Grad, Deconv and LRP we blur their maps by a Gaussian kernel with $\sigma = 0.02 \cdot \max\{W, H\}$, which slightly improves their performance since their maps tend to be sparse and noisy at the pixel level. In the evaluation, we expand the ground truth region by a tolerance margin of 15 pixels, so that the attention maps produced by CAM, which are only 7 pixels in the shortest dimension, can be more fairly compared.

6.1.2 Results

The results are reported in Table 1. Our c-MWP consistently achieves competitive performance on both VOC07 and COCO across different CNN models. In addition, c-MWP is also substantially better than MWP, which validates the idea of contrastive attention.

The performance of Grad and Deconv varies with different CNN architectures. For example, Grad attains competitive performance with CNN-S on VOC07, but it gives significantly worse performance than other methods with ResNet50. Deconv attains comparable performance to Grad with VGG16 and is much better than Grad with ResNet50, but it is inferior to Grad on the other architectures. Both Grad and Deconv give significantly worse performance than c-MWP with ResNet50. In particular, on the difficult subsets of VOC07 and COCO, c-MWP outperforms Grad and Deconv by nearly 30 points. This indicates that the gradient based methods do not generalize well on different model architectures.

[2] https://github.com/jimmie33/Caffe-ExcitationBP.

[3] https://github.com/BVLC/caffe/wiki/Model-Zoo.

[4] On COCO, we need to compute about 116K attention maps, which leads to over 950 h of computation on a single machine for LRP using VGG16.

Table 1 Mean accuracy (%) in the Pointing Game

	VOC07 Test (All/Diff.)				COCO Val. (All/Diff.)			
	CNN-S	VGG16	GoogleNet	ResNet50	CNN-S	VGG16	GoogleNet	ResNet50
Center	69.5/42.6	69.5/42.6	69.5/42.6	69.5/42.6	27.7/19.4	27.7/19.4	27.7/19.4	27.7/19.4
Grad	78.6/59.8	76.0/56.8	79.3/61.4	65.8/50.9	38.7/30.1	37.1/30.7	42.6/36.3	30.4/24.9
Deconv	73.1/45.9	75.5/52.8	74.3/49.4	73.0/53.0	36.4/28.4	38.6/30.8	35.7/27.9	38.2/31.2
LRP	68.1/41.3	–	72.8/50.2	–	32.5/24.0	–	40.2/32.7	–
CAM	–	–	80.8/61.9	**90.6/81.8**	–	–	41.6/35.0	**58.4/53.5**
MWP	73.7/52.9	76.9/55.1	79.3/60.4	80.9/66.4	35.0/27.7	39.5/32.5	43.6/37.1	46.8/40.9
c-MWP	**78.7/61.7**	**80.0/66.8**	**85.1/72.3**	89.2/81.3	**43.0/37.0**	**49.6/44.2**	**53.8/48.3**	57.4/52.5

For each method, we report two scores for the overall test set and a difficult subset respectively. `Center` is the baseline that points at image center. The best scores are shown in bold in each column and the second best scores are underlined

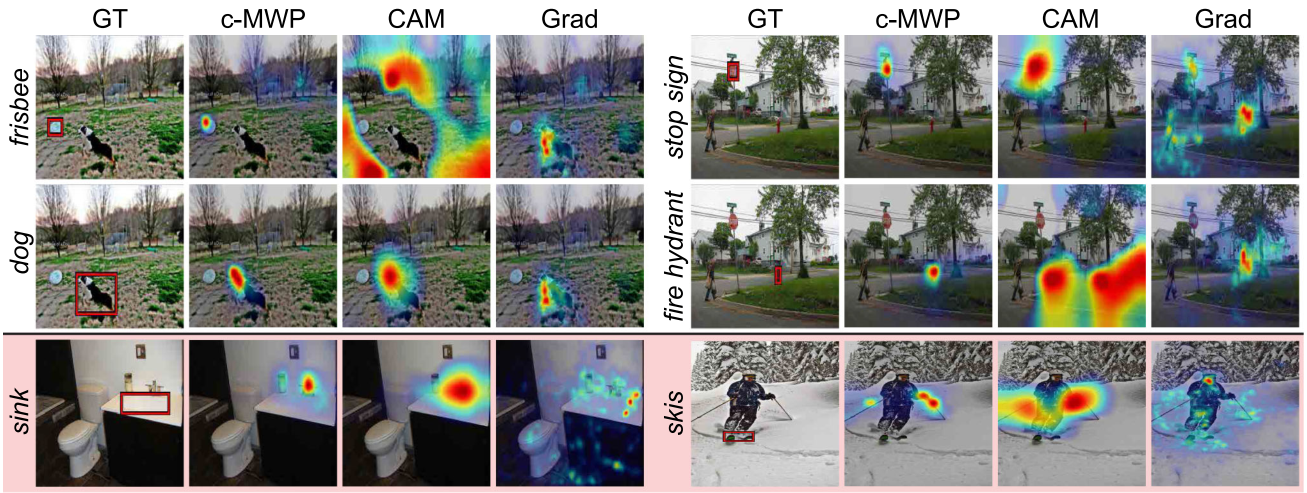


Fig. 5 Example attention maps using GoogleNet. For visualization, the maps are superimposed on the images after some postprocessing (slight blur for Grad and thresholding for CAM). (Top two rows) Our c-MWP is very discriminative and can often localize challengingly small objects like `frisbee`, `stop sign` and `fire hydrant`. (Bottom row) Two typical failure cases of top-down neural attention are shown. Since `faucet` often co-occurs with `sink`, the CNN's attention falsely focuses on the faucet in the image. It is the same case for `ski poles` and `skis`

Models like GoogleNet and ResNet50 use the Average Pooling layer instead of the Fully-connected layer to generate global image features. This type of CNN model provides the best localization performance for our method. With GoogleNet, our c-MWP outperforms the second best method CAM by over 10 points on the difficult subsets of VOC07 and COCO. In particular, we find that our c-MWP gives the best performance in 69/80 object categories of COCO, especially for small objects like `remote`, `tie` and `baseball bat`. A detailed figure of the per-class results is presented in the supplementary material. With ResNet50, c-MWP is about 1 point worse than CAM on VOC07 and COCO. We will provide more analysis in the following section to explain the effect of model architectures on the performance of c-MWP.

In general, the proposed c-MWP gives leading performance in the Pointing Game and shows much better generalizability than other compared methods to different CNN architectures.

Example attention maps are shown in Fig. 5 (more sample results can be found in the supplementary material). As we can see, our c-MWP maps can accurately localize the cued objects in rather challenging scenes. Two typical failure cases of top-down neural attention are also shown in Fig. 5. For example, since `faucet` often co-occurs with `sink`, the CNN's attention falsely focuses on the faucet in the image. It is the same case for `ski poles` and `skis`.

6.1.3 Analysis

We now provide analysis of our method using the Pointing Game.

Layer Selection Effects For the Pointing Game, we use GoogleNet to analyze the effects of layer selection. For a comparison, we also report the performance of Grad and Deconv by taking the maximum gradient magnitude across

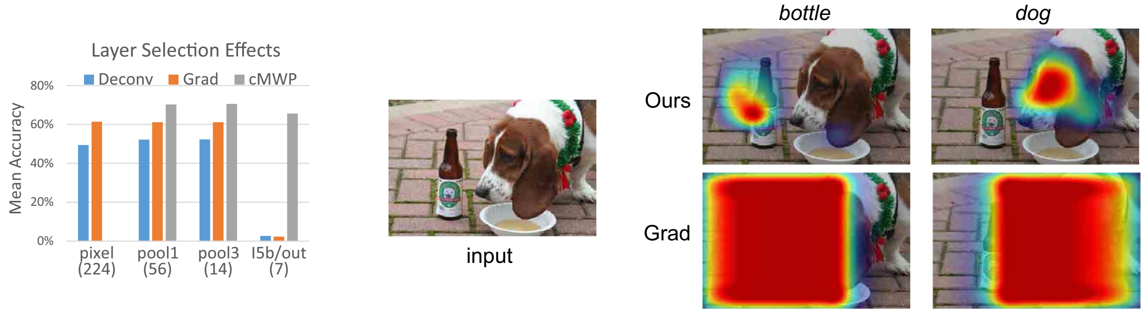


Fig. 6 Effects of layer selection on VOC07 *difficult* set. (Left) For Grad, Deconv and our c-MWP, we compare their attention maps from three different layers in the GoogleNet. At I5b/out, Grad and Deconv fail to generate meaningful attention maps, while our method can still achieve reasonable accuracy. Since the proposed Excitation Backprop is not well defined at the pixel level (see Sect. 3.2), we omit its performance evaluation in that category. (Right) We show example attention maps by our c-MWP and Grad from the I5b/out layer

Table 2 CAM versus c-MWP*

	VOC07 Test (All/Diff.)		COCO Val. (All/Diff.)	
	GoogleNet	ResNet50	GoogleNet	ResNet50
CAM	80.8/61.9	90.6/81.8	41.6/35.0	58.4/53.5
c-MWP*	80.7/65.6	89.1/80.9	41.5/34.8	57.0/52.0
c-MWP	85.1/72.3	89.2/81.3	53.8/48.3	57.4/52.5

We compare the performance of CAM and c-MWP* with GoogleNet and ResNet50, where c-MWP* denote the contrastive attention maps generated from the layer right before the Average Pooling layer (i.e. the Inception_5b/output layer for GoogleNet and the res5c layer for ResNet50). In other words, the output layer of c-MWP* is the same layer as used by CAM (i.e. the layer before the Average Pooling layer). c-MWP* and CAM give similar performance. We also include the performance of the original c-MWP reported in Table 1

feature map channels in the intermediate layers. Results are reported in Fig. 6. We choose three intermediate layers in GoogleNet: pool1, pool3 and Inception_5b/output (I5b/out), whose spatial resolutions are 56, 14 and 7 in the shortest dimension respectively. Note that since the proposed Excitation Backprop is not defined at the pixel level (see Sect. 3.2), we omit its performance evaluation in that category. As demonstrated, the effect of the layer selection on our method is quite marginal in the Pointing Game when the spatial resolution of the selected layer is about 14 × 14 or above. Our c-MWP only gets a slight decrease in accuracy (mainly due to the map's low spatial resolution), while Grad and Deconv do not generate meaningful attention maps (see Fig. 6). This is because the attention maps of Grad and Deconv at I5b/out are not conditioned on the activation values of I5b/out, and thus fail to leverage the spatial information captured by I5b/out.

CAM and contrastive MWP As discussed in Sect. 4.1, when the Average Pooling layer is used to generate the global image feature for the final classification layer, the proposed c-MWP method and the CAM method can produce very similar results. In Table 2, we empirically verify this assumption. We show the performance of c-MWP*, which uses the layer right before the Average Pooling layer to compute the attention maps, i.e. the Inception_5b/output layer for GoogleNet and the res5c layer for ResNet50. As we can see, c-MWP* and CAM are very close in performance for both GoogleNet and ResNet50.

Compared with the original c-MWP reported in Table 1, c-MWP* produces attention maps of lower resolution (7 pixels in the shortest dimension). For GoogleNet, this leads to a significant performance gap between the original c-MWP and c-MWP*. For example, c-MWP outperforms c-MWP* by over 10 points with GoogleNet on the COCO dataset. With ResNet50, however, there is little performance difference between c-MWP and c-MWP*. To see why, we show an example attention map of c-MWP with ResNet50 in Fig. 7b. As we can see, the map computed using a lower level layer (res3a) has a very clear strided pattern. This is mainly due to two factors. First, ResNet50 uses strided convolution instead of max pooling to down-sample the feature maps. More specifically, ResNet50 uses 1 × 1 convolution filters with a stride of two for down-sampling, which basically down-samples the feature maps using a regular spatial grid. This of course will generate strided patterns when the top-down signals are passed through the strided convolution layers. Second, if most of the top-down signals pass through the skip connections of the residual units, the regular patterns will be kept in lower layers. We demonstrate in Fig. 7c what happens if we remove the skip connections during Excitation Backprop. As we can see, without passing the top-down signals through the skip connections, the strided patterns disappear in lower layers. However, we find that ignoring the skip connections does not improve the localization accuracy of c-MWP.

Analysis of Contrastive Top-Down Attention The proposed contrastive attention is conceptually simple: one attention map is subtracted from its reverse map using the virtual

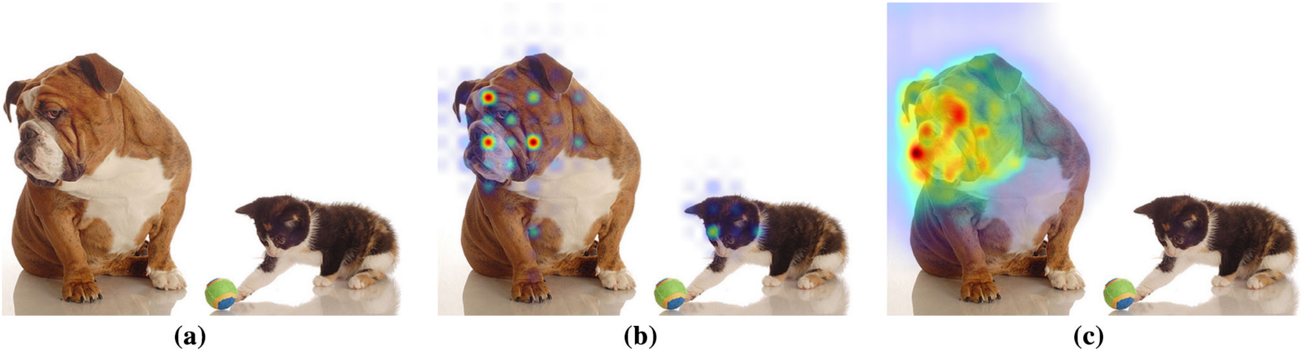


Fig. 7 Strided pattern in attention maps from ResNet50. **a** a test image; **b** the c-MWP map of dog using the res3a layer of ResNet50; **c** the c-MWP map of dog generated by ignoring the skip connections during Excitation Backprop. The strided pattern disappears when the skip connections are not used for backpropagation

Table 3 Analysis of contrastive attention on VOC07 *difficult* set using GoogleNet

	Excitation backprop			Other methods		
	Full	Post-norm	w/o norm	c-Grad	c-Deconv	c-CAM
Mean acc. (%)	**70.6**	58.1	41.6	N.A.	67.7 (49.4)	61.9 (61.9)

We evaluate two variants of Excitation Backprop for the contrastive attention map computation compared with our full model. We also test the contrastive attention idea for Grad, Deconv and CAM and their original scores are shown in brackets. See text for details

The best score is shown in bold

contrastive output unit. We test this idea for Grad, Deconv and CAM and the performance is reported in Table 3. For Grad, the gradient magnitude map is identical to its reverse map since the gradients of the reverse map are just the negation of the reference map. As a result, the subtraction gives a zero map. For CAM, the performance remains the same because the reverse map is again a negation of the reference attention map and the maximum point will not be changed by the subtraction. However, the proposed contrastive attention works for Deconv, when the attention map and its reverse map are L1-normalized before subtraction. Deconv shares a similar spirit of our method as it discards negative/inhibitatory signals by thresholding at ReLU layers, but it also introduces non-linearity in the propagation process. Therefore, it requires two backward passes and proper normalization, while our method can directly propagate the contrastive signal via a single pass and achieves better performance.

Our probabilistic WTA formulation produces well-normalized attention maps that enable direct subtraction. We report the performance of two variants of our method in Table 3. We remove the normalization factor Z_i in Eq. 6 and pass down the contrastive signal. This leads to a significant degradation in performance (w/o norm). Then we compute the attention map and its reverse map separately and do the subtraction after L1-normalization (post-norm). The performance is improved but still substantially lower than our full method. This analysis further confirms the importance of our probabilistic formulation.

Table 4 Deterministic WTA versus probabilistic WTA (MWP and c-MWP) on VOC07 (All/Diff.) using GoogleNet

	d-WTA	MWP	c-MWP
Mean acc. (%)	72.3/49.7	79.3/60.4	**85.1/72.3**

The best score is shown in bold

Probabilistic Versus deterministic WTA We further provide some analysis regarding the comparison between the deterministic WTA (d-WTA) and our proposed probabilistic WTA (p-WTA). As described above, d-WTA can only generate very sparse binary maps, due to its greedy selection of the winner neurons. In contrast, our p-WTA generates soft maps, which is a crucial advantage, especially for computing contrastive attention maps.

We evaluate the d-WTA in the Pointing Game using the GoogleNet model, and the result is reported in Table 4. As we can see, d-WTA gives comparable performance to LRP and Deconv, but it is substantially worse than both MWP and c-MWP.

Table 5 Bounding box localization error on ImagNet Val. using GoogleNet

	Grad	Deconv	LRP	CAM	Feedback[a]	c-MWP	MWP
Opt. α	5.0	4.5	1.0	1.0	–	0.0	1.5
Loc. Error (%)	41.6	41.6	57.8	48.1	38.8	57.0	**38.7**

[a] The score of Feedback is from the original paper

The best score is shown in bold and the second best score is underlined

6.2 Localizing Dominant Objects

We now turn to a different evaluation setting used by Cao et al. (2015). The goal of this setting is bounding box localization of dominant objects in the image.

6.2.1 Experiment Setting

We follow the protocol of Feedback Net (Cao et al. 2015) for a fair comparison. The test is performed on the ImageNet Val. set (∼50K images), where each image has a label representing the category of dominant objects in it. The label is given, so the evaluation is based on the localization error rate with an IOU threshold at 0.5. Images are resized to 224×224.

As in Cao et al. (2015), simple thresholding is used to extract a bounding box from an attention map. We set the threshold $\tau = \alpha\mu_I$, where μ_I is the mean value of the map. Then the tightest bounding box covering the white pixels is extracted. The parameter α is optimized in the range [0:0.5:10] for each method on a held out set.

6.2.2 Results

Table 5 reports the results based on the same GoogleNet model obtained from Caffe Model Zoo as in Cao et al. (2015). We find that c-MWP performs poorly, but our MWP obtains competitive results against Feedback and other methods. Compared with Feedback, our method is conceptually much simpler. Feedback requires modification of a CNN's architecture and needs 10-50 iterations of forward-backward passes for computing an attention map.

Note that this task favors attention maps that fully cover the *dominant* object in an image. Thus, it is very different from the Pointing Game, which favors discriminativeness instead. Our c-MWP usually only highlights the most discriminative part of an object due to the competition between the contrastive pair of top-down signals. This experiment highlights the versatility of our method, and the value of the non-contrastive version (MWP) for dominant object localization. Some visual results are shown in Fig. 8.

6.3 Text-to-Region Association

Text-to-region association in unconstrained images (Plummer et al. 2015) is very challenging compared to the object detection task, due to the lack of fully-annotated datasets and the large number of words/phrases used in the natural language. Moreover, an image region can be referred to by potentially many different words/phrases, which further increases the complexity of the fully-supervised approach.

By leveraging the top-down attention of a CNN image tag classifier, we propose a highly scalable approach to weakly supervised word-to-region association. We train an image tag classifier using ∼6M weakly labeled thumbnail images collected from a commercial stock image website[5] (Stock6M). Each image is 200-pixels in the longest dimension and comes with about 30–50 user tags. These tags cover a wide range of concepts, including objects, scenes, body parts, attributes, activities, and abstract concepts, but are also very noisy. We picked ∼18K most frequent tags for our dictionary. We empirically found that the first few tags of each image are usually more relevant, and consequently use only the first 5 tags of an image in the training.

6.3.1 Experiment Setting

Tag Classifier Training We use the pre-trained GoogleNet model from Caffe Model Zoo, and fine-tune the model using the multi-label cross-entropy objective function for the 18K tags. Images are padded to square shape by mirror padding and upsampled to 256×256. Random flipping and cropping are used for data augmentation. We use SGD with a batch size of 64 and a starting learning rate of 0.01. The learning rate is lowered by a factor of 0.1 when the validation error plateaus. The training process passes through the data for three epochs and takes ∼55 h on an NVIDIA K40c GPU.

Dataset and Evaluation To quantitatively evaluate our top-down attention method and the baselines in text-to-region association, we use the recently proposed Flickr30k Entities (Flickr30k) dataset (Plummer et al. 2015). Evaluation is performed on the test split of Flickr30k (1000 images), where every image has five sentential descriptions. Each Noun Phrase (NP) in a sentence is manually associated with the bounding box regions it refers to in the image. NPs are grouped into eight types (see Plummer et al. (2015)). Given a NP, the task is to provide a list of scored bounding boxes, which will be measured by the recall rate (similar to the object

[5] https://stock.adobe.com.

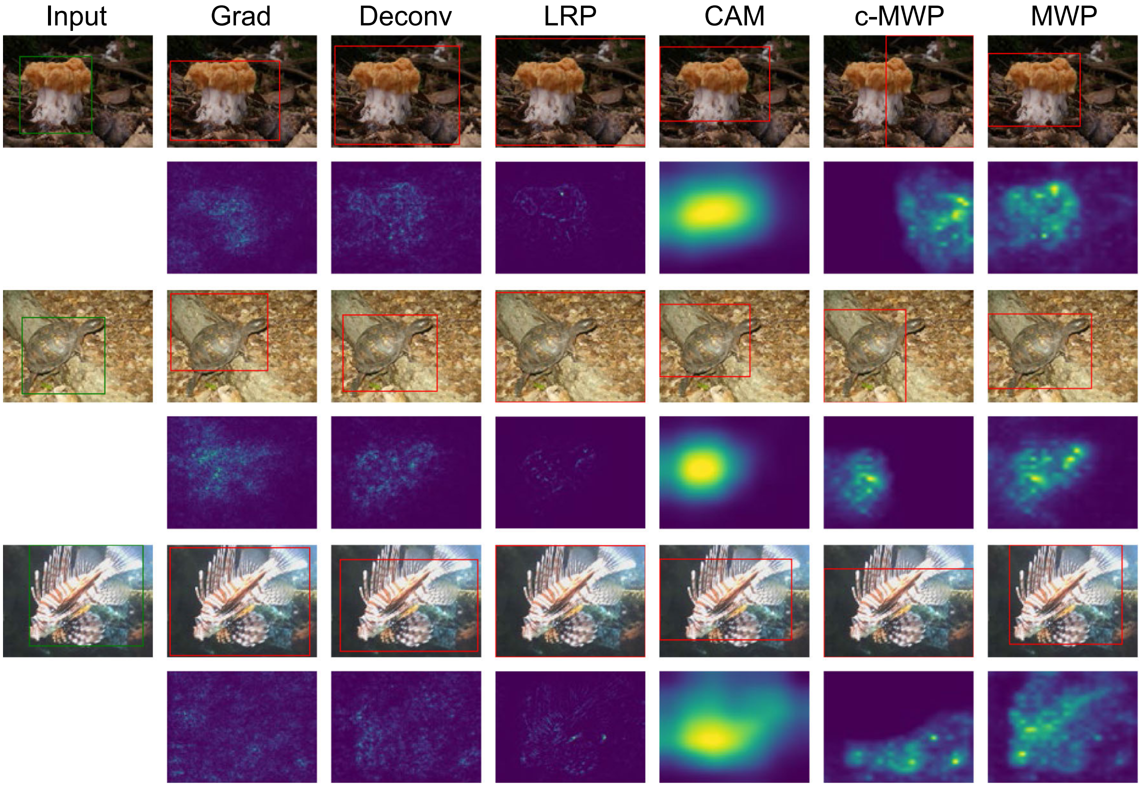


Fig. 8 Example results of compared methods for localizing dominant objects on ImageNet. The first column shows the input images with the ground truth bounding box. The following columns show the bounding box prediction of different methods as well as their corresponding attention maps. LRP maps often have small noise on the image boundaries, leading to big boxes. Increasing the map threshold to remove boundary noise will shrink the boxes but also decrease the scores

proposal metric) or per-group/per-phrase Average Precision (AP) (similar to the object detection metric). We use the evaluation code from Plummer et al. (2015).

To generate scored bounding boxes for a NP, we first compute the word attention map for each word in the NP using our tag classifier. Images are resized to 300 pixels in the shortest dimension to better localize small objects. Then we simply average the word attention maps to get a NP attention map. Advanced language models can be used for better fusing the word attention maps, but we adopt the simplest fusion scheme to demonstrate the effectiveness of our top-down attention model. We skip a small proportion of words that are not covered by our 18K dictionary. MCG (Arbeláez et al. 2014) is used to generate 500 segment proposals, which are re-scored based on the phrase attention map. The re-scored segments are then converted to bounding boxes, and redundant bounding boxes are removed via Non-maximum Suppression using the IOU threshold of 0.7.

The segment scoring function is defined as $f(R) = S_R/A_R^\gamma$ where S_R is the sum of the values inside the segment proposal R on the given attention map and A_R is the segment's area. The parameter γ is to control the penalty of the segment's area, which is optimized for each method in the range [0:0.25:1].

6.3.2 Results

The recall rates and mAP scores are reported in Table 6. For our method and the baselines, we additionally report the performance on a subset of small instances whose bounding box area is below 0.25 of the image size, as we find small regions are much more difficult to localize. Our c-MWP consistently outperforms all the attention map baselines across different metrics. In particular, the group-level mAP of our method is better than the second best by a large margin.

We also compare with a recent fully supervised method (Plummer et al. 2015), which is trained directly on the Flickr30k Entities dataset using CNN features. For fair comparison, we use the same bounding box proposals used in Plummer et al. (2015), which are generated by Edge-Boxes (EB) (Zitnick and Dollár 2014). These proposals are pre-computed and provided by Plummer et al. (2015). Our performance using EB is lower than using MCG, mainly due to the lower accuracy of the EB's bounding box proposals. Compared with the segmentation proposals, the bounding

Table 6 Performance comparison on the Flickr30k Entities dataset

	opt. γ	R@1	R@5	R@10	mAP (Group)	mAP (Phrase)
MCG_base	–	10.7/7.7	30.3/22.4	40.5/30.3	6.9/4.5	16.8/12.9
Grad (MCG)	0.50	24.3/7.6	49.6/32.9	59.7/45.8	10.2/3.8	28.8/15.6
Deconv (MCG)	0.50	21.5/11.3	48.4/34.5	58.5/46.0	10.0/4.0	26.5/16.7
LRP (MCG)	0.50	24.3/11.8	51.6/36.8	61.3/48.5	10.3/4.3	28.9/18.1
CAM (MCG)	0.75	21.7/6.5	47.1/27.9	56.1/39.1	7.5/2.0	26.0/11.9
MWP (MCG)	0.50	**28.5**/15.0	52.7/39.1	61.3/49.8	11.8/5.3	**31.1**/20.3
c-MWP (MCG)	0.50	26.2/**21.2**	**54.3/43.4**	**62.2/51.7**	**15.2/10.8**	30.8/**24.0**
CCA* (EB)	–	25.2/**21.8**	**50.3/41.0**	58.1/**47.3**	12.8/**11.5**	28.8/**23.6**
CCA (EB)	–	25.3/–	–	**59.7**/–	11.2/–	–
c-MWP (EB)	0.25	**27.0**/18.4	49.9/35.2	57.7/43.9	**13.2**/8.1	**29.4**/20.0

We report performance for both the whole dataset and a subset of small instances. The R@N refers to the overall recall rate regardless of phrase types. mAP (Group) and mAP (Phrase) should be interpreted differently, because most phrases belong to the group `people`. CCA* refers to the precomputed results provided by Plummer et al. (2015), while CCA is the results reported in the original paper. MCG_base is the performance using MCG's original proposal scores. EB is EdgeBoxes (Zitnick and Dollár 2014)

The best scores are shown in bold

Table 7 Per group recall@5 (%) on the Flickr30k Entities dataset

	People	Clothing	Bodypart	Animal	Vehicle	Instrument	Scene	Other	Mean
MCG_base	36.1	30.1	9.9	50.8	37.8	26.5	31.5	19.1	30.3
Grad (MCG)	65.0	32.4	14.0	**70.1**	63.0	40.7	58.8	32.5	47.1
Deconv (MCG)	65.4	31.6	18.7	67.0	64.0	46.9	53.6	28.9	47.0
LRN (MCG)	64.6	37.7	16.4	62.9	63.5	45.7	59.4	37.9	48.5
CAM (MCG)	60.5	28.4	9.6	57.0	57.5	37.0	**64.4**	32.7	43.4
MWP (MCG)	**68.6**	37.7	16.1	68.7	66.3	53.7	54.5	36.8	50.3
c-MWP (MCG)	63.5	**47.6**	**24.5**	69.9	**72.0**	**54.3**	61.0	**40.2**	**54.1**
CCA* (EB)	**63.6**	**43.7**	**22.9**	57.0	69.0	50.6	45.0	**36.2**	48.5
c-MWP (EB)	62.8	35.0	17.6	**65.1**	**73.5**	**58.6**	**53.2**	**36.2**	**50.3**

The mean scores are computed over different group types, which are different from the overall recall rates reported in Table 6

The best scores are shown in bold

box proposals can also affect our ranking function for small and thin objects. However, our method still attains competitive performance against Plummer et al. (2015). Note that our method is weakly supervised and does not use any training data from the Flickr30k Entities dataset.

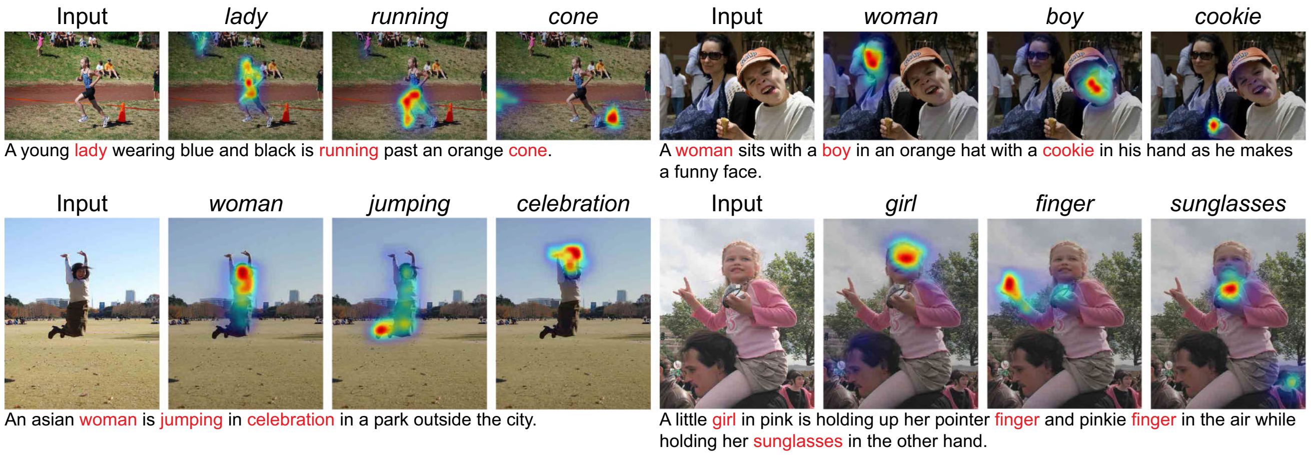
We further report the per-group Recall@5 score in Table 7. Our method achieves promising results in many group types, e.g. `vehicle` and `instrument`. Note that the fully supervised CCA (EB) (Plummer et al. 2015) gives significantly worse performance than c-MWP (EB) in `animal`, `vehicle` and `instrument`, which are the three rarest types in the Flickr30k Entities dataset. This again shows the limitation of fully-supervised approaches due to the lack of fully-annotated data.

Some example word attention maps are shown in Fig. 9 to demonstrate the localization ability of our method. As we can see, our method can localize not only noun phrases but also actions verbs in the text. More results can be found in the supplementary material.

7 Example Applications in Model Interpretation and Data Annotation

In this section we discuss applications of our top-down neural attention method in model interpretation and data annotation. In tasks like medical image analysis or fine-grained image classification, grounding a neural network model's prediction can not only help users better understand the mechanism and capability of the model, but also provide guidance for data annotation. We provide two examples of such applications of our top-down neural attention method.

Facial Expression Analysis A VGG-S model is trained to classify a face image into one of the six basic facial emotions Levi



A young lady wearing blue and black is running past an orange cone.

A woman sits with a boy in an orange hat with a cookie in his hand as he makes a funny face.

An asian woman is jumping in celebration in a park outside the city.

A little girl in pink is holding up her pointer finger and pinkie finger in the air while holding her sunglasses in the other hand.

Fig. 9 Word attention maps obtained by c-MWP using our image tag classifier. For each test image, one of its caption annotations from Flickr30k Entities is displayed below. We show the attention maps for the words in red in each caption. By leveraging a large-scale weakly labeled dataset, our method can localize a large number of visual concepts, e.g. objects (cone, sunglasses and cookie), fine-grain categories of people (woman and boy), body parts (finger) and actions (jumping, running and celebration)

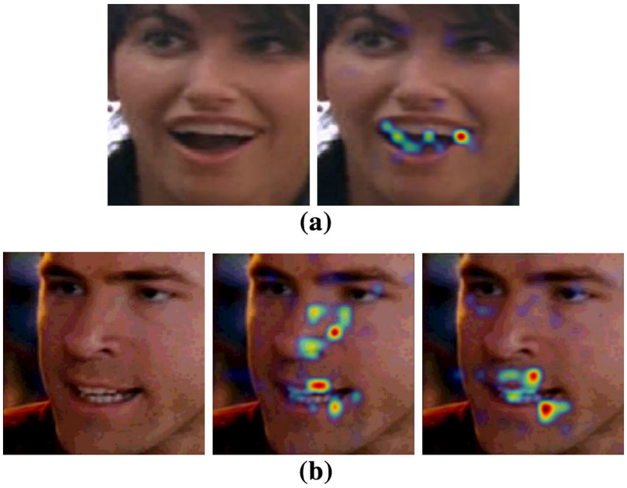


Fig. 10 This figure shows **a** a correctly classified example ("Happy"), together with the evidence the model uses for the classification, an **b** an incorrectly classified example (left) with a ground truth label "Angry". This image is mis-classified by the trained model as "Happy". We demonstrate using c-MWP why the network thinks this is "Happy" (right). We also show the c-MWP map of "Angry" (middle)

and Hassner (2015) (Angry, Happy, Sad, Surprise, Disgust, Fear), and Neutral. This model is trained on the training set of the Static Facial Expressions in the Wild (SFEW) dataset Dhall et al. (2012).

Using the proposed c-MWP maps, we can visualize local evidence used by our model for a target emotion category. Sample analysis of correct and incorrect classifications of the model using validation images from the SFEW dataset are demonstrated in Fig. 10. In Fig. 10a, the c-MWP for the correctly classified category "Happy" shows that the model uses the evidence around the mouth. In Fig. 10b, the c-MWP maps for "Angry" (ground truth) and "Happy" (model's prediction) give different focused regions. It indicates that for "Angry", the model is looking for evidence around the nose and eyes in addition to the mouth. This type of visualization can also be useful for human annotator to annotate facial action units (e.g. Chin Raiser, Nose Wrinkler, Outer Brow Raiser, Jaw Drop) under the Facial Action Coding System.[6] Annotating such facial action units requires training. The top-down attention maps can help non-professional annotators localize action units that are relevant to a high-level emotion.

Medical Image Analysis Medical image analysis tasks are usually quite demanding. Machine learning methods can help speedup these tasks, but in many scenarios it still requires human experts to examine the predicted results. Thus, it is of special importance that machine learning models can point human experts to the relevant regions that may support or reject the predicted results.

Huang et al. (2017) use a neural network model to label fetal heart orientation on ultrasound images. To verify their model learns the task-specific features, they use Excitation Backprop to produce attention maps for their model (See Fig. 11).

Jamaludin et al. (2017) propose a neural network model to automatically produce radiological gradings of spinal lumbar MRIs. They demonstrate that the c-MWP maps of their model generated by our method can clearly localize pathological

[6] The Facial Action Coding System (FACS) is a taxonomy for encoding facial muscle movements into Action Units (AUs). Combinations of coded action units are used to make higher-level decisions, such as a facial emotion: happy, sad, angry, etc.

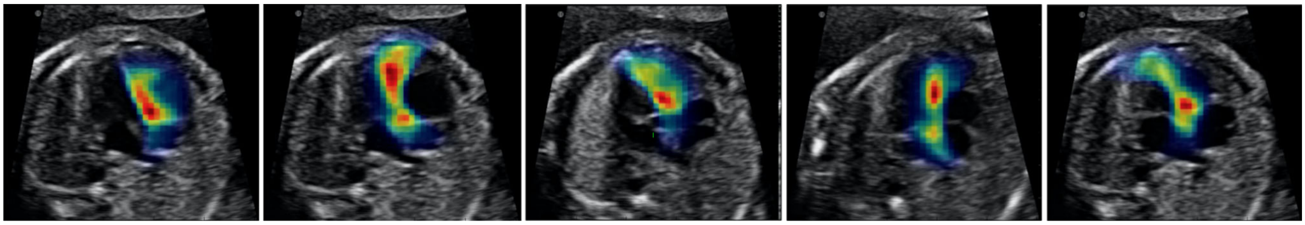


Fig. 11 Image courtesy of Huang et al. (2017). It shows that the neural network model proposed by Huang et al. (2017) knows the key location that defines the fetal heart orientation on ultrasound images

regions in the disc volumes, even although no segmentation annotation is used during their model training.

These applications indicate that the proposed top-down neural attention formulation can be a powerful tool for computer-aided image analysis and annotation.

8 Conclusion

We propose a probabilistic Winner-Take-All formulation to model the top-down neural attention for CNN classifiers. Based on our formulation, a novel propagation method, Excitation Backprop, is presented to compute the Marginal Winning Probability of each neuron. Using Excitation Backprop, highly discriminative attention maps can be efficiently computed by propagating a pair of contrastive top-down signals via a single backward pass in the network. A theoretic connection is drawn between our contrastive top-down attention formulation and the Class Activation Map computation. In experiments, we demonstrate the accuracy and the generalizability of our method in a large-scale Pointing Game. Our method gives leading performance in the Pointing Game with four different CNN models. We further show the usefulness of our method in localizing dominant objects. Moreover, without using any localization supervision or language model, our neural attention based method attains competitive localization performance versus a recent fully supervised method on the challenging Flickr30k Entities dataset. We also discuss the applicability of Excitation Backprop for analytical and annotational purposes.

Acknowledgements This research was supported in part by Adobe Research, US NSF Grants 0910908 and 1029430, and gifts from NVIDIA.

References

Anderson, C. H., & Van Essen, D. C. (1987). Shifter circuits: A computational strategy for dynamic aspects of visual processing. *Proceedings of the National Academy of Sciences*, *84*(17), 6297–6301.

Arbeláez, P., Pont-Tuset, J., Barron, J., Marques, F., & Malik, J. (2014). Multiscale combinatorial grouping. In *CVPR*.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS ONE*, *10*(7), e0130140.

Baluch, F., & Itti, L. (2011). Mechanisms of top-down attention. *Trends in Neurosciences*, *34*(4), 210–224.

Bazzani, L., Bergamo, A., Anguelov, D. & Torresani, L. (2016). Self-taught object localization with deep networks. In *2016 IEEE winter conference on applications of computer vision (WACV)* (pp. 1–9). IEEE.

Beck, D. M., & Kastner, S. (2009). Top-down and bottom-up mechanisms in biasing competition in the human brain. *Vision Research*, *49*(10), 1154–1165.

Cao, C., Liu, X., Yang, Y., Yu, Y., Wang, J., Wang, Z., et al. (2015). Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *ICCV*.

Chatfield, K., Simonyan, K., Vedaldi, A. & Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*.

Clevert, D.-A., Unterthiner, T., & Hochreiter, S. (2016). Fast and accurate deep network learning by exponential linear units (elus). In *ICLR*.

Desimone, R. (1998). Visual attention mediated by biased competition in extrastriate visual cortex. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, *353*(1373), 1245–1255.

Desimone, R., & Duncan, J. (1995). Neural mechanisms of selective visual attention. *Annual Review of Neuroscience*, *18*(1), 193–222.

Dhall, A., Goecke, R., Lucey, S., & Gedeon, T. (2012). Collecting large, richly annotated facial-expression databases from movies. *IEEE MultiMedia*, *19*(3), 34–41.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, *88*(2), 303–338.

Fang, H., Gupta, S., Iandola, F., Srivastava, R. K., Deng, L., Dollár, P., et al. (2015). From captions to visual concepts and back. In *CVPR*.

Fong, R., & Vedaldi, A. (2017). Interpretable explanations of black boxes by meaningful perturbation. arXiv:1704.03296.

Gonzalez-Garcia, A., Modolo, D., & Ferrari, V. (2016). Do semantic parts emerge in convolutional neural networks? arXiv:1607.03738.

Guillaumin, M., Küttel, D., & Ferrari, V. (2014). Imagenet auto-annotation with segmentation propagation. *International Journal of Computer Vision*, *110*(3), 328–348.

He, K., Zhang, X., Ren, S. & Sun, J. (2016). Deep residual learning for image recognition. In *CVPR* (pp. 770–778).

Huang, W., Bridge, C. P., Noble, J. A.,& Zisserman, A. (2017). Temporal heartnet: Towards human-level automatic analysis of fetal cardiac screening video. arXiv:1707.00665.

Jamaludin, A., Kadir, T., & Zisserman, A. (2017). Spinenet: Automated classification and evidence visualization in spinal mris. *Medical Image Analysis*, *41*, 63–73.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., et al. (2014). Caffe: Convolutional architecture for fast feature embedding. In *ACM international conference on multimedia*.

Kemeny, J. G., Snell, J. L., et al. (1960). *Finite Markov chains*. New York: Springer.

Koch, C., & Ullman, S. (1987). Shifts in selective visual attention: Towards the underlying neural circuitry. In L. M. Vaina (Ed.), *Matters of intelligence. Synthese library (Studies in epistemology, logic, methodology, and philosophy of science)* (vol 188, pp. 115–141). Dordrecht: Springer.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *NIPS*.

Levi, G., & Hassner, T. (2015). Emotion recognition in the wild via convolutional neural networks and mapped binary patterns. In *Proceedings of the 2015 ACM on international conference on multimodal interaction* (pp. 503–510). ACM.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). Microsoft COCO: Common objects in context. In *ECCV*.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *CVPR* (pp. 3431–3440).

Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2015). Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *CVPR*.

Papandreou, G., Chen, L.-C., Murphy, K., & Yuille, A. L. (2015). Weakly-and semi-supervised learning of a dcnn for semantic image segmentation. In *ICCV*.

Pathak, D., Krahenbuhl, P., & Darrell, T. (2015). Constrained convolutional neural networks for weakly supervised segmentation. In *ICCV*.

Pinheiro, P. O., & Collobert, R. (2014). Recurrent convolutional neural networks for scene parsing. In *ICLR*.

Pinheiro, P. O., & Collobert, R. (2015). From image-level to pixel-level labeling with convolutional networks. In *CVPR*.

Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., & Lazebnik, S. (2015). Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *CVPR*.

Reynolds, J. H., & Heeger, D. J. (2009). The normalization model of attention. *Neuron*, *61*(2), 168–185.

Russakovsky, O., Deng, J., Hao, S., Krause, J., Satheesh, S., Ma, S., et al. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, *115*(3), 211–252. https://doi.org/10.1007/s11263-015-0816-y.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & LeCun, Y. (2014). Overfeat: Integrated recognition, localization and detection using convolutional networks. In *ICLR*.

Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR workshop*.

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *ICLR*.

Springenberg, J. T., Dosovitskiy, A., Brox, T., & Riedmiller, M. (2014). Striving for simplicity: The all convolutional net.arXiv preprint. arXiv:1412.6806.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2015). Going deeper with convolutions. In *CVPR*.

Treisman, A. M., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, *12*(1), 97–136.

Tsotsos, J. K., Culhane, S. M., Wai, W. Y. K., Lai, Y., Davis, N., & Nuflo, F. (1995). Modeling visual attention via selective tuning. *Artificial Intelligence*, *78*(1), 507–545.

Usher, M., & Niebur, E. (1996). Modeling the temporal dynamics of it neurons in visual search: A mechanism for top-down selective attention. *Journal of Cognitive Neuroscience*, *8*(4), 311–327.

Wolfe, J. M. (1994). Guided search 2.0 a revised model of visual search. *Psychonomic Bulletin and Review*, *1*(2), 202–238.

Wolfe, J. M., Butcher, S. J., Lee, C., & Hyle, M. (2003). Changing your mind: On the contributions of top-down and bottom-up guidance in visual search for feature singletons. *Journal of Experimental Psychology: Human Perception and Performance*, *29*(2), 483.

Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., & Lipson, H. (2015). Understanding neural networks through deep visualization. arXiv:1506.06579

Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In *ECCV*.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2015). Object detectors emerge in deep scene cnns. In *ICLR*.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2016). Learning deep features for discriminative localization. In *CVPR*.

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. (2014). Learning deep features for scene recognition using places database. In *NIPS*.

Zitnick, C. L., & Dollár, P. (2014). Edge boxes: Locating object proposals from edges. In *ECCV*.