EPiC Series in Computing

Volume 41, 2016, Pages 72–80

GCAI 2016. 2nd Global Conference on Artificial Intelligence

Capacity Visual Attention Networks

Marcus Edel and Joscha Lausch

Freie Universität Berlin, Berlin, Germany
{marcus.edel, jo.lausch}@fu-berlin.de

Abstract

Inspired by recent work in machine translation and object detection, we introduce an attention-based model that automatically learns to extract information from an image by adaptively assigning its capacity across different portions of the input data and only processing the selected regions of different sizes at high resolution. This is achieved by combining two modules: an attention sub-network which uses a mechanism to model a human-like counting process and a capacity sub-network. This sub-network efficiently identifies input regions for which the attention model output is most sensitive and to which we should devote more capacity and dynamically adapt the size of the region. We focus our evaluation on the Cluttered MNIST, SVHN, and Cluttered GTSRB image datasets. Our findings indicate that the proposed model is able to drastically reduce the number of computations, compared with traditional convolutional neural networks, while maintaining similar or better performance.

1 Introduction

Deep Neural Networks (DNNs) have substantially pushed Artificial Intelligence in a wide range of tasks, including but not limited to object recognition from images [13, 25], speech recognition [21, 30] and even Atari and Go games [17, 1].

Although DNNs are extending the state of the art in the last decade, they are almost exclusively trained on one or many very fast and power-hungry Graphic Processing Units or on industrial-sized clusters [27, 5]. So, while they perform well on expensive, GPU-based machines or industrial-sized clusters, they are often unsuitable for smaller devices like cell phones and embedded systems. For example, AlexNet [13] has 61M parameters (249MB of memory) and performs 1.5B high precision operations to make a prediction. These numbers are even higher for bigger networks with more parameters that call for more resources (processing power, memory, battery time, etc), which are often critical constraints.

The inefficient use of resources is often based on the assumption, that all input regions contain the same amount of information. Indeed, convolutional neural networks apply the same set of filters uniformly across the complete input, while recurrent neural networks (RNNs) apply the same transformation at every time step. Those networks lead to time-consuming training and prediction, because they require a large number of multiplications.

However, the relevant information is often not uniformly distributed across the input data. For example, objects in images are spatially localized, i.e. a traffic light is often located in the upper image region. This observation has been recently exploited in attention-based systems

[16], which can reduce computations significantly by learning to selectively attend to relevant input regions.

In addition to the assumption that not all regions contain the same amount of information, another challenge is the dimensionality of the structured output, which is bounded by the number of pixels multiplied by the maximum number of objects. Previous attention-based models will have trouble to dynamically adapt the size of the focus region, which may result in loss of information.

To tackle both these challenges, we propose a new model based on a recurrent neural network that utilizes visual attention to perform object recognition. Rather than using a sliding window approach that searches over the entire image, attention allows for salient features to dynamically be foregrounded as needed. This is especially important when there is a lot of clutter in an image. Unfortunately, the attention based approach has one potential drawback of losing information, by using a static focus size. We address this issue, by using a recurrent capacity-network, which is able to dynamically adapt the size of the focused image region. In particular, we make the following contributions:

- We introduce an attention-based model that is able to recognize objects in an input image. Since the model is non-differentiable, it can be trained using reinforcement learning procedure as used in [16].
- We introduce a recurrent sub-network, that is capable to dynamically adapt the capacity of the focus region, which we believe is crucial for large-scale image recognition tasks.
- We conduct three sets of experiments, which show that it is possible to train and to evaluate a capacity attention model that achieves nearly state-of-the-art results.
- We explore the parameter space in various experiments and explain some of the apparent inertia in the proposed capacity attention model.

2 Related Work

Several methods have been proposed for recognizing and classifying objects in an image. Many of these methods are based on convolutional neural networks and inspired by the successful use of neural networks built on top of engineered features like SIFT [15]. One major reason why convolutional neural networks are well suited [13, 25, 21] is the availability of large annotated datasets and fast GPU computing [27, 5], and also due to some important methodological developments such as dropout regularization [23], rectifier linear activations [18] and improved optimizer functions [26].

Spurred by the recent success in a wide range of tasks [22, 13, 25, 21], several contributions have been dedicated to reducing the cost of the widespread sliding window paradigm or to reduce the overall model architecture. Lampert et al. [14] proposed a branch-and-bound scheme to find the highest scored window while evaluating the classifier as few times as possible over all candidate regions. However, it is restricted to classifiers for which a good upper bound on a set of windows exists. Other works [28, 20] used a shallow network representation to reduce the number of activation and weight multiplications, by exploiting the underlying network structure, to imitate a much larger network model. The authors in [4, 6, 11] use context to improve object detection and recognition. They employed background-to-object context to avoid false-positive detections, or used relations between multiple objects [4, 6]. These methods are different from our approach because they use context as an additional cue on top of the detector, whereas

our approach uses an attention mechanism that helps the model to direct its focus only on important input regions.

Closest to our work are techniques that consider attention mechanisms to capture visual structure with biologically inspired, foveation-like methods [16, 2]. In particular, our work extends the recurrent attention model (RAM) proposed in [16]. While this model has been shown to learn successful strategies to learn various image data sets, it only uses a number of static glimpse sizes. In our approach, we use an additional sub-network, to dynamically change the glimpse size, with the assumption to increase the performance. Moreover, we use reinforcement learning to tackle the general structured prediction problems, rather than an end-to-end backpropagation approach.

3 Capacity Visual Attention Model

The basic structure of the Capacity Visual Attention model (CRAM) is similar to that of the RAM model [16]: A glimpse network captures salient information about the input image at a specific position and region size; a classification network uses the information to condition its prediction of the input regions. However, there are two key differences. Firstly, a dynamically updated attention mechanism is used to restrict both the input region observed by the glimpse network, and the next output region prediction from the emission network. In simple terms, the sub-network decides at each time-step what the capacity of the focus region should be. Secondly, the outputs of the capacity sub-network are successively added to the input of the emission network that will ultimately generate the information for the next focus region. Allowing the emission network to combine the information from the location and the capacity networks, instead of just using the information independently. The architecture is illustrated in Fig. 1.

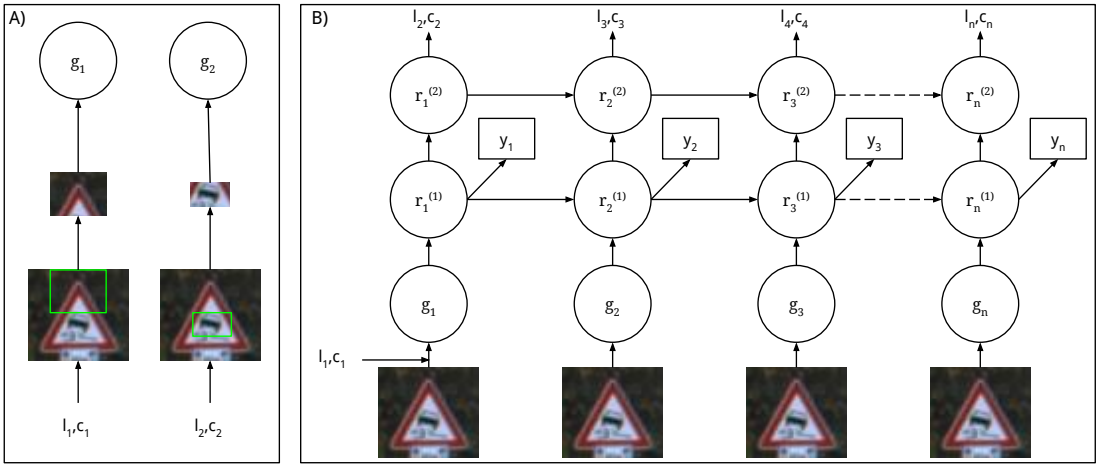

Figure 1: Conventional Capacity Visual Attention Model. At each time-step a location l_n and capacity size c_n is passed to the glimpse network, which then generates the representative feature vector g_n on the basis of the current image. During inference the result is passed to the location and capacity recurrent network. The RNNs at the previous time-step specifies where the attention shifts to. The combined output of the lower recurrent layer $r_n^{(1)}$ is used to compute the approximate posterior to generate the output.

3.1 Architecture

Our proposed model can be broken down into four major sub-network, each network maps some input into a vector output. In particular, at each step n, the model receives a location l_n and a focus region size c_n along with the raw image. The model uses the information to update its internal state and to direct the next location and focus size at the next time-step.

Glimpse network: The glimpse network is a trainable non-linear function, that incorporates the information from the raw image $image$, the location $l_n = (x_n, y_n)$ and the focus size $c_n = (w_n, h_n)$ into a single vector g_n. Following [3] we denote $G_{image}(\cdot)$ which returns a parameterized image region, using the current image region x_n and it's parameters W_{image}. Additionaly, we denote $G_{location}$, which returns a parameterized representation of the current location l_n using the location parmeters $W_{location}$. Separately the cacpacity tuple is mapped by $G_{capacity}(c_n|W_{image})$ and $G_{capacity}(c_n|W_{location})$ repsectively. To get the final feature vector g_n, we multipy the vectors $G_{image}(\cdot)$ and $G_{location}(\cdot)$ element-wise.

$$\mathcal{W}_{image} = G_{capacity}(c_n|\mathcal{W}_{image}) \tag{1}$$
$$\mathcal{W}_{location} = G_{capacity}(c_n|\mathcal{W}_{location}) \tag{2}$$
$$g_n = G_{image}(x_n|\mathcal{W}_{image}) \cdot G_{location}(l_n|\mathcal{W}_{location}) \tag{3}$$

Location network: The location network aggregates information from the individual glimpses and combines the information in a coherent manner. The job of the location network is to preserve spatial information by using the given input to the location network at each time step, to build the basis for the prediction of the next location. The location network consists of two recurrent layers with non-linear function. We use the Long Short-Term Memory architecture [12] in the extended form with forget gates [7] for the location network. We favour LSTM due to its proven track record for handling long-range dependencies in real sequential data [8, 7].

$$rl_n^{(1)} = RL(g_n, rl_{n-1}^{(1)}|\mathcal{W}_{rl^1}) \; and \; rl_n^{(2)} = RL(rl_n^{(1)}, rl_{n-1}^{(2)}|\mathcal{W}_{rl^2}) \tag{4}$$

Capacity network: The Capacity network is similar to the location network, it also aggregates information from the individual glimpses and combines the information in a coherent manner. With the key difference that it uses the preserve spatial information to build the basis for the prediction of the next focus size. The capacity network consists of two recurrent layers with non-linear function. We use the same architecture, as for the location network.

$$rc_n^{(1)} = RC(g_n, rc_{n-1}^{(1)}|\mathcal{W}_{rc^1}) \; and \; rc_n^{(2)} = RC(rc_n^{(1)}, rc_{n-1}^{(2)}|\mathcal{W}_{rc^2}) \tag{5}$$

Emission network: The emission network takes the current state history encapsulated by the hidden state of the location and capacity networks and makes a decision on the next location and size of the focus region. Its job is to incorporate the location and capacity information as well as past information, to direct the attention.

$$l_{n+1} = E(rl_{n^(2)}|\mathcal{W}_{emission}) \; and \; c_{n+1} = E(rc_{n^(2)}|\mathcal{W}_{emission}) \tag{6}$$

Classification network: The classification network outputs a prediction for the class label y based on the lower recurrent layers of the location and capacity networks. The classification network has one fully connected hidden layer and a softmax output layer for the class y.

$$P(y|I) = O(rl_n^1, rc_n^1|\mathcal{W}_{output}) \tag{7}$$

3.2 Training

Following Mnih et al. [16], we trained the classification and glimpse network by backpropagating the error through the networks where the REINFORCE algorithm [29] was used for the non-differentiable attention based location and capacity network. The REINFORCE algorithm is designed to train stochastic units, conditioned on an input, by adjusting the parameters of the agent (objective function) used in the training process. Unlike backpropagation, the agent doesn't have to be differentiable.

4 Experiments

In this section we explore the use of the Capacity Attention Model on a number of supervised learning tasks. Experiments on Cluttered MNIST and Cluttered GTSRB, shows the ability of the model to improve classification performance by actively adapting the size of the focus region.

4.1 Model Specification

In the experiments we trained a Capacity Neural Attention Model end-to-end, where we learn recurrent and fully connected layers jointly. Details of the network architectures can be found in section 3.

4.2 Datasets

We performed experiments on several datasets, for every dataset, the network's goal is to recognize the objects accordingly. The details for each dataset used in our experiments are listed below.

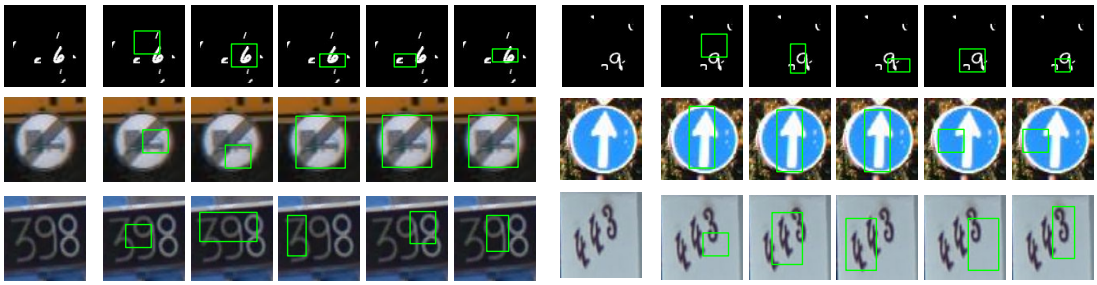

Figure 2: Two examples of the learned policy for each dataset. The first column shows the input image while the next 5 columns show the selected glimpse locations and focus size. (1. row: Cluttered MNIST, 2. row: Cluttered GTSRB, 3. row: SVHN)

Cluttered MNIST: We use the MNIST based Cluttered MNIST digit classification dataset as proposed by Mnih et al. in [16]. Each image in this dataset is a hand-written MNIST digit

located randomly on a 100 x 100 black image with random 8 x 8 subpatches sampled from other random MNIST digits. Since the dataset is based on the MNIST dataset it has 60000 images for training and 10000 for testing.

SVHN: The SVHN [19] is a real-world image dataset, obtained from house numbers in Google Street View images. The task of the network is to recognize the digit sequence, which can be of length 1 to 5 digits. The dataset has three subsets: train (33k), extra (202k) and test (13k). We trained the model on 230k image using both the train and extra subsets. We used the rest of the train and extra subset for choosing the hyperparameters and the test subset for testing.

Cluttered GTSRB: The most challenging aspects of recognizing and classifiying objects in real world images is the presence of a wide range of clutter or noise. To reflect this real world scenario, we created a Cluttered GTSRB dataset based on the Cluttered MNIST idea. The data for this dataset was generated by placing a traffic sign from the GTSRB dataset [24] in a random location of a larger 100 x 100 black image, with random 8 x 8 subpatches sampled from random GTSRB traffic sign images. Since the image sizes in the dataset vary between 15 x 15 to 250 x 250 we normalized each image to a uniform image size of 15 x 15.

Fig. 2 shows random samples of test cases for the Cluttered MNIST, SVHN and Cluttered GTSRB dataset.

4.3 Baselines

As baselines we use the Recurrent Visual Attention Models (RAM) as poposed by Mnih et al. [16] and DRAW as proposed by Gregor et al. [10]. To compare our results with a traditional model, we also implemented a deep convolutional neural network (CNN) with a similar number of parameters as the other models. The network has 8 convolutional layers with 128 filters in each followed by 2 fully connected layers of 3096 rectifier units. We also used Dropout with 50% dropout rate to prevent over-fitting.

4.4 Evaluation Results

	Cluttered MNIST	**SVHN**	**Cluttered GTSRB**
RAM	10.35 %	-	15.85 %
DRAW	5.13 %	20.10 %	8.35 %
CNN	13.31 %	40.55 %	18.20 %
CRAM	**4.90 %**	**18.90 %**	**8.10 %**

Table 1: Classification test error on the Cluttered MNIST, SVHN and Cluttered GTSRB dataset.

Table 1 shows the results on the Cluttered MNIST, SVHN, and Cluttered GTSRB dataset. The model proposed by [10] has an error rate of 20.10% on the SVHN dataset, while by using our proposed model, we decrease the error rate to 18.90%.

The results confirm that the additional capacity sub-network can dynamically adapt the size of the focus region, and is able to focus on relevant information that is distributed across the input image. In addition, it also verifies our assumption, that our overall model is able to learn to track objects despite the presence of background noise and distractor objects, despite

its limited bandwidth sensor. Therefore the proposed model is able to outperform the other baseline models in terms of classification accuracy. Figure 2 shows samples of the selected patches by the attention mechanism and the different region sizes as chosen by the capacity sub-network.

parameter (millions)	**10 layer CNN**	**CRAM avg.**
Cluttered MNIST (100x100)	169	20
SVHN (32x32)	55	20
Cluttered GTSRB (100x100)	169	20

Table 2: Computation cost of Capacity Visual Attention Networks (CRAM) V.S. the implemented CNN.

Similar to the observations about the invariance against noise and distractor objects, it seems that our model is able to outperform the attention model that uses a glimpse network with different static scales at each time step, as suggested by Mnih et al. in [16]. This is partly because the model often appeares to follow the objects mean location while changing the sensor capacity to capture the object in its attention. Another observation is that the attention-based models nearly always scale better (in terms of computation cost) than the other CNN based algorithms. Table 2 shows that our algorithm satisfies its original goal: to be able to scale effectively with the image size and number of objects.

5 Conclusion and Discussion

We have presented Capacity Visual Attention Networks, a novel attention based approach for object recognition and classification. We showed that this model can learn to solve a number of recognition problems and generalize well to problems that incorporate noise or distracting objects, by adaptively assigning its capacity across different portions of the input data. Our model achieved state-of-the-art performance on the cluttered MNIST and cluttered GTSRB classification dataset. It is favorable over traditional convolutional networks because it has a small memory footprint and may be used in conjunction with data approximation schemes for additional speedup. There are still interesting future directions to pursue. The first direction is to train the model end-to-end with backpropagation instead of using a policy network trained by reinforcement learning. The second direction is to use another memory representation in favor of the LSTM based approach, such as an adapted version of the recent proposed Neural Turing Machines [9].

References

[1] Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, jan 2016.

[2] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *CoRR*, abs/1412.7755, 2014.

[3] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *CoRR*, abs/1412.7755, 2014.

[4] M. J. Choi, J. J. Lim, A. Torralba, and A. S. Willsky. Exploiting hierarchical context on a large database of object categories. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 129–136, June 2010.

[5] Adam Coates, Brody Huval, Tao Wang, David J. Wu, Bryan C. Catanzaro, and Andrew Y. Ng. Deep learning with COTS HPC systems. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pages 1337–1345, 2013.

[6] Chaitanya Desai, Deva Ramanan, and Charless C. Fowlkes. Discriminative models for multi-class object layout. *Int. J. Comput. Vision*, 95(1):1–12, October 2011.

[7] F A Gers, J Schmidhuber, and F Cummins. Learning to forget: continual prediction with LSTM. *Neural computation*, 12(10):2451–2471, 2000.

[8] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610, 2005.

[9] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *CoRR*, abs/1410.5401, 2014.

[10] Karol Gregor, Ivo Danihelka, Alex Graves, and Daan Wierstra. DRAW: A recurrent neural network for image generation. *CoRR*, abs/1502.04623, 2015.

[11] Geremy Heitz and Daphne Koller. Learning spatial context: Using stuff to find things. In *Computer Vision - ECCV 2008, 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I*, pages 30–43, 2008.

[12] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[14] Christoph H. Lampert, Matthew B. Blaschko, and Thomas Hofmann. Efficient subwindow search: A branch and bound framework for object localization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(12):2129–2142, December 2009.

[15] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, November 2004.

[16] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. *CoRR*, 2014.

[17] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602, 2013.

[18] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 807–814, 2010.

[19] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011.

[20] S. Oniga and J. Sütö. Human activity recognition using neural networks. In *Control Conference (ICCC), 2014 15th International Carpathian*, pages 403–406, May 2014.

[21] Tara N. Sainath, Brian Kingsbury, Abdel-rahman Mohamed, George E. Dahl, George Saon, Hagen Soltau, Tomas Beran, Aleksandr Y. Aravkin, and Bhuvana Ramabhadran. Improvements to deep convolutional neural networks for lvcsr. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 315–320, 2013.

[22] P. Sermanet and Y. LeCun. Traffic sign recognition with multi-scale convolutional networks. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pages 2809–2813, July 2011.

[23] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*,

15(1):1929–1958, January 2014.

[24] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In *IEEE International Joint Conference on Neural Networks*, pages 1453–1460, 2011.

[25] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014.

[26] T. Tieleman and G. Hinton. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

[27] Vincent Vanhoucke, Andrew Senior, and Mark Z. Mao. Improving the speed of neural networks on cpus. In *Deep Learning and Unsupervised Feature Learning Workshop, NIPS 2011*, 2011.

[28] Keze Wang, Xiaolong Wang, Liang Lin, Meng Wang, and Wangmeng Zuo. 3d human activity recognition with reconfigurable convolutional neural networks. *CoRR*, abs/1501.06262, 2015.

[29] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3-4):229–256, May 1992.

[30] Dong Yu and Li Deng. *Automatic Speech Recognition - A Deep Learning Approach*. Springer, October 2014.