LEARN TO PAY ATTENTION

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose an end-to-end-trainable attention module for convolutional neural network (CNN) architectures built for image classification. The module takes as input the 2D feature vector maps which form the intermediate representations of the input image at different stages in the CNN pipeline, and outputs a 2D matrix of scores for each map. Standard CNN architectures are modified through the incorporation of this module, and trained under the constraint that a convex combination of the intermediate 2D feature vectors, as parametrised by the score matrices, must *alone* be used for classification. Incentivised to amplify the relevant and suppress the irrelevant or misleading, the scores thus assume the role of attention values. Our experimental observations provide clear evidence to this effect: the learned attention maps neatly highlight the regions of interest while suppressing background clutter. Consequently, the proposed function is able to bootstrap standard CNN architectures for the task of image classification, demonstrating superior generalisation over 6 unseen benchmark datasets. When binarised, our attention maps outperform other CNN-based attention maps, traditional saliency maps, and top object proposals for weakly supervised segmentation as demonstrated on the Object Discovery dataset. We also demonstrate improved robustness against the fast gradient sign method of adversarial attack.

1 INTRODUCTION

Feed-forward convolutional neural networks (CNNs) have demonstrated impressive results on a wide variety of visual tasks, such as image classification, captioning, segmentation, and object detection. However, the visual reasoning which they implement in solving these problems remains largely inscrutable, impeding understanding of their successes and failures alike.

One approach to visualising and interpreting the inner workings of CNNs is the attention map: a scalar matrix representing the relative importance of layer activations at different 2D spatial locations with respect to the target task (Simonyan et al., 2013). This notion of a nonuniform spatial distribution of relevant features being used to form a task-specific representation, and the explicit scalar representation of their relative relevance, is what we term 'attention'. Previous works have shown that for a classification CNN trained using image-level annotations alone, extracting the attention map provides a straightforward way of determining the location of the object of interest (Cao et al., 2015; Zhou et al., 2016) and/or its segmentation mask (Simonyan et al., 2013), as well as helping to identify discriminative visual properties across classes (Zhou et al., 2016). More recently, it has also been shown that training smaller networks to mimic the attention maps of larger and higher-performing network architectures can lead to gains in classification accuracy by those smaller networks (Zagoruyko & Komodakis, 2016).

The works of Simonyan et al. (2013); Cao et al. (2015); Zhou et al. (2016) form a progressive series of increasingly sophisticated techniques for estimating attention maps in CNNs. However, these approaches share a crucial limitation: all are implemented as post-hoc additions to fully trained networks. Further, using these attention maps to crop regions for reclassification in the manner of Cao et al. (2015) splits the pipeline into two or more independently tuned steps, meaning that there is no joint optimisation with respect to the final classification loss.

Thus, for the first time in the context of classification CNNs, we propose a probabilistic attention estimator, embedded within the network architecture, whose parameters are learned over the course of end-to-end training of the entire network. Our efforts are motivated in part by recent improvements

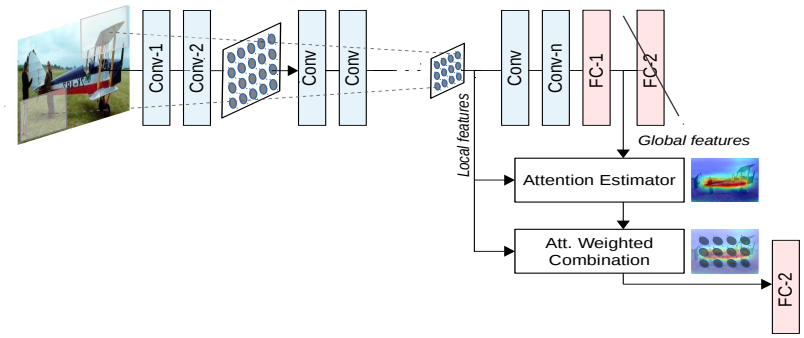

Figure 1: Overview of the proposed attention mechanism.

in the tasks of machine translation (Bahdanau et al., 2014) and image caption generation (Xu et al., 2015) via the use of attention in the context of RNNs. Note that while Mnih et al. (2014) and Jaderberg et al. (2015) integrate the learning of their respective attention representations into their training pipelines, they both model attention non-probabilistically. Our attention matrix is probabilistic in its representation of attention. This difference becomes crucial in cases where higher-order probabilistic reasoning over the attention maps is sought, e.g. when applying conditional random fields to refine attention maps for weakly supervised segmentation. The matrix is itself a deterministic function of the input image.

Fig. 1 provides an overview of the proposed method. Henceforth, we will use the terms 'local features' and 'global features' to refer to features extracted by some layer of the CNN whose effective receptive fields are, respectively, contiguous proper subsets of the image ('local') and the entire image ('global'). By defining a compatibility measure between local and global features, we redesign standard architectures such that they must classify the input image using *only* a weighted combination of local features, with the weights represented here by the attention map. The network is thus forced to learn a pattern of attention relevant to solving the task at hand.

We experiment with applying the proposed attention mechanism to the popular CNN architectures of VGGNet (Simonyan & Zisserman, 2014) and ResNet (He et al., 2015), and capturing coarse-to-fine attention maps at multiple levels. We observe that the proposed mechanism can bootstrap baseline CNN architectures for the task of image classification: for example, adding attention to the VGG model offers an accuracy gain of 7% on CIFAR-100. Our use of attention-weighted representations leads to improved fine-grained recognition and superior generalisation on 6 benchmark datasets for domain-shifted classification. As observed on models trained for fine-grained bird recognition, attention aware models offer limited resistance to adversarial fooling at low and moderate L_∞-noise norms. The trained attention maps outperform other CNN-derived attention maps (Zhou et al., 2016), traditional saliency maps (Jiang et al. (2013); Zhang & Sclaroff (2013)), and top object proposals on the task of weakly supervised segmentation of the Object Discovery dataset (Rubinstein et al. (2013)). In §5, we present sample results which suggest that these improvements may owe to the method's tendency to highlight the object of interest while suppressing background clutter.

2 Related Work

Attention in deep classification networks: Early approaches to understanding the performance of CNNs as stacked feature hierarchies focus on capturing the image semantics that activate specific features, particularly at higher-up network layers (Girshick et al., 2014; Simonyan et al., 2013; Donahue et al., 2014). For example, Girshick et al. (2014) cluster the image patches that activate features in a given CNN layer: this enables visualisation of the patterns learned by the network, but does not identify the visual support that the network relies upon in making a classification decision. Simonyan et al. (2013) approximate CNNs as linear functions, interpreting the gradient of a class

output score with respect to the input image as that class's spatial support in the image domain, i.e. its attention map. Importantly, they are one of the first to successfully demonstrate the use of attention for localising objects of interest using image-level category labels alone. Zhou et al. (2016) apply the classifier weights learned for image-level descriptors to patch descriptors, and the resulting class scores are used as a proxy for attention. Their improved localisation performance comes at the cost of classification accuracy. Cao et al. (2015) introduce attention in the form of binary nodes between the network layers of Simonyan et al. (2013). While the attention maps are adapted to the end task, these nodes require additional parameter tuning at test time. Notably, all of the above methods extract attention from fully trained CNN classification models, i.e. in a post-hoc manner. Again, we demonstrate the classification and generalisation advantages of training an attention mechanism that is built into a CNN pipeline.

Robustness to adversarial attacks: The work by Goodfellow et al. (2014) explores the ease of fooling the current generation of deep networks by adding an imperceptible perturbation to the input image in the direction of the locally estimated normal to the class decision boundary. This suggests that the sample images, at least in some dimensions, are situated dangerously close to the decision hyperplanes. Wang et al. (2016) argue that this weakness emerges from the reliance on spurious non-causal image features for classification which makes the network vulnerable along these extra dimensions. Our experiments confirm this to the extent that when parts of the input image, and hence the corresponding features, may be selectively ignored we can achieve some increased robustness to adversarial attacks.

Tackling domain shift: CNNs are successful at the task of object recognition on experimental datasets such as PascalVOC (Everingham et al., 2015), where the train and test data distributions are extremely similar to one another. However, real-world images are challenging as they include wide variations in object pose, background content, resolution, occlusion, and illumination conditions. As these variations are not necessarily fully represented by the experimental datasets used to train networks, successfully applying those networks in the real world involves addressing the domain shift problem. A traditional approach to handling domain shift in CNNs involves fine-tuning on the new dataset, which may require thousands of images from each target category for successful adaptation. Past studies have also explored domain adaptation by feature transformation, a process that requires sampling the target distribution to approximate the data transformation from the source to the target. Here, we show that attention-aware models can offer significant benefits in handling domain changes, particularly those involving background content, occlusion, and object pose variation, by selectively attending to the object of interest. In a manner similar to Girshick et al. (2014) and Donahue et al. (2014), we compare how well out-of-the-box CNN features from attention-aware models and their standard counterparts are able to classify previously unseen datasets.

Weakly supervised semantic segmentation: Some of the most effective approaches to semantic segmentation require expensive annotation of thousands of images per category (Everingham et al., 2015). To counter this, weakly supervised methods leverage scribbles, bounding boxes and image-level category labels for image segmentation. We demonstrate the use of an attention mechanism built into standard CNN architectures to obtain segmentations as a by-product of training for the task of image classification. We compare with other works in this area including those that use traditional definitions of saliency (Jiang et al., 2013; Zhang & Sclaroff, 2013), object proposal techniques (Arbeláez et al., 2014) and joint-inference for co-segmentation (Joulin et al., 2010; 2012; Chen et al., 2014; Dutt Jain & Grauman, 2016) for the task of weakly supervised semantic segmentation.

3 Approach

The core goal of this work is to use attention maps to identify and exploit the effective spatial support of the information used by CNNs in making their classification decisions. This approach is premised on the hypothesis that there is benefit to identifying salient regions and amplifying their influence, while likewise suppressing the irrelevant and potentially confusing information in other regions. In particular, we expect that enforcing a more focused and parsimonious use of information over image space should aid in generalisation over changes in the data distribution, as occurs for instance when training on one set and testing on another. Thus, we propose a trainable attention estimator and illustrate how to integrate it into standard CNN pipelines so as to influence their output as intended. The method is based on enforcing a notion of compatibility between local feature vectors

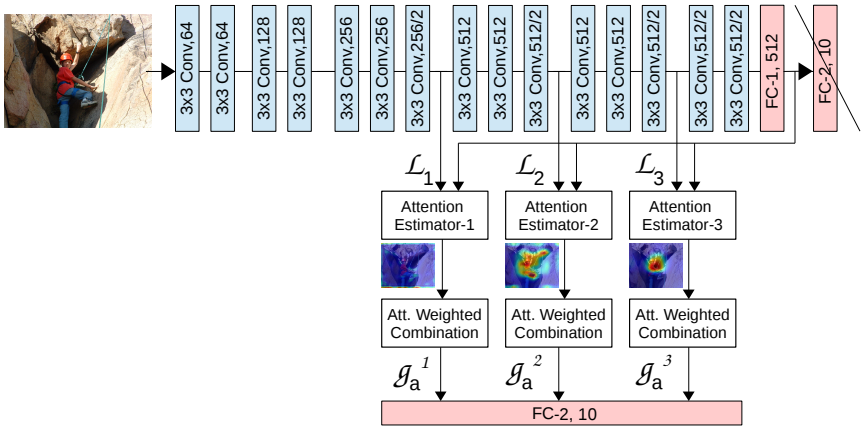

Figure 2: Attention introduced at 3 distinct layers of VGG. Lowest level attention maps appear to focus on the surroundings (*i.e.*, the rocky mountain), intermediate level maps on object parts (*i.e.*, harness and climbing equipment) and the highest level maps on the central object.

extracted at intermediate stages in the CNN pipeline and the global feature vector normally fed to the linear classification layers at the end of the pipeline. We implement attention-aware classification by restricting the classifier to use *only* a collection of local feature vectors, as chosen and weighted by the compatibility scores, in classifying examples. We will first discuss the modification to the network architecture and the method of training it, given a choice of compatibility score. We will then conclude the method description by presenting alternate choices of the score.

3.1 Design and training of attention submodule

The proposed approach is illustrated in Fig. 1. Denote by $\mathcal{L}^s = \{\ell_1^s, \ell_2^s, \cdots, \ell_n^s\}$ the set of feature vectors extracted at a given convolutional layer denoted by $s \in \{1, \cdots, S\}$, where each ℓ_i^s is the vector of output activations at the spatial location i of n total spatial locations in the layer. g is the global feature vector which has the entire input image as support and is output by the network's series of convolutional and nonlinear layers, having only to pass through the final fully connected layers to produce the original architecture's class score for that input. Assume for now the existence of a compatibility function $\mathcal{C}$ which takes two vectors of equal dimension as arguments and outputs a scalar compatibility score: this will be specified in 3.2.

The method proceeds by computing, for each of one or more layers s, the set of compatibility scores $\mathcal{C}(\hat{\mathcal{L}}^s, g) = \{c_1^s, c_2^s, ...c_n^s\}$, where $\hat{\mathcal{L}}^s$ is the image of $\mathcal{L}^s$ under a linear mapping of the ℓ_i^s to the dimensionality of g. The compatibility scores are then normalised by a softmax operation:

$$a_i^s = \frac{\exp(c_i^s)}{\sum_j^n \exp(c_j^s)}, \; i \in \{1 \cdots n\}. \tag{1}$$

The normalised compatibility scores $\mathcal{A}^s = \{a_1^s, a_2^s, \cdots a_n^s\}$ are then used to produce a single vector $g_a^s = \sum_{i=1}^n a_i^s \cdot \ell_i^s$ for each layer s, by simple element-wise weighted averaging. Crucially, the g_a^s now *replace* g as the global descriptor(s) for the image. For a network trained under the restriction that the g_a^s alone are used to classify the input image, $\mathcal{A}$ corresponds to 'attention' as we have defined it earlier.

In the case of a single layer ($S = 1$), the attention-incorporating global vector g_a is computed as described above, then mapped onto a T-dimensional vector which is passed through a softmax layer to obtain class prediction probabilities $\{\hat{p}_1, \hat{p}_2, \cdots \hat{p}_T\}$, where T is the number of target classes. In the case of multiple layers ($S > 1$), we compare two options: concatenating the global vectors into a single vector $g_a = [g_a^1, g_a^2, \cdots g_a^S]$ and using this as the input to the linear classification step as above, or, using S different linear classifiers and averaging the output class probabilities. All free network parameters are learned in end-to-end training under a cross-entropy loss function.

3.2 CHOICE OF COMPATIBILITY FUNCTION $\mathcal{C}$

The compatibility score function $\mathcal{C}$ can be defined in various ways. The alignment model from Bahdanau et al. (2014); Xu et al. (2015) can be re-purposed as a compatibility function as follows:

$$c_i = \langle u, \ell_i + g \rangle,\ i \in \{1 \cdots n\}, \tag{2}$$

Given the free parameters between the local and the global image descriptors in a CNN pipeline, we simplify the concatenation of the two descriptors to an addition operation and learn a single fully connected mapping to the compatibility scores. Here, the weight vector u can be interpreted as learning the universal set of features relevant to the object categories in the dataset. In that sense, the weights may be understood as learning the general concept of objectness.

Alternatively, we can use the dot product between g and ℓ_i as a measure of their compatibility:

$$c_i = \langle \ell_i, g \rangle,\ i \in \{1 \cdots n\}. \tag{3}$$

In this case, the relative magnitude of the scores would depend on the alignment between g and ℓ_i in the high dimensional feature space and the strength of activation of ℓ_i.

3.3 INTUITION

In a standard CNN architecture, a global image descriptor g is derived from the input image and passed through a fully connected layer to obtain class prediction probabilities. The network must express g via mapping of the input into a high-dimensional space in which salient higher-order visual concepts are represented by different dimensions, so as to render the classes linearly separable from one another. Our method encourages the filters *earlier* in the CNN pipeline to learn mappings compatible with the mapping that produces g in the original architecture, by allowing a local descriptor ℓ_i of an image patch to contribute to the final classification step only in proportion to its compatibility with g as detailed above. That is, $\mathcal{C}(\hat{\ell}_i, g)$ should be high if and only if the corresponding patch contains parts of the dominant image category. Note that this implies that the effective filters operating over image patches in the layers s must represent relatively 'mature' features with respect to the classification goal. We thus expect to see the greatest benefit in deploying attention relatively late in the pipeline. Note also that the use of the softmax function in normalising the compatibility scores enforces $0 \leq a_i \leq 1\ \forall i \in \{1 \cdots n\}$ and $\sum_i a_i = 1$, i.e. that the combination of feature vectors is convex. This ensures that features at different spatial locations must effectively compete against one another for their share of the attention map. The compatibility scores thus serve as a robust proxy for attention in the classification pipeline.

4 EXPERIMENTAL SETUP

Baselines and proposed architectures: For our attention model, we remove the first 2 max-pooling layers of the VGG network and place them after two additional convolutional layers at the end of the pipeline. This ensures that the local layers used for estimating attention are reasonably sized. Thus our modified model has 17 layers - 15 convolutional and 2 fully-connected layers. The output activations of layer-16 (FC layer) define our global feature vector g. We use g to experiment with the proposed attention framework at layers 7, 10 and 13 (Conv layers). We contrast our approach with the modification proposed by Zhou et al. (2016), namely VGG-GAP. For ResNets (He et al., 2015), we use a 164-layered network. We replace the spatial average-pooling step after the computational block 4 by convolutional and max-pooling steps which gives us our global feature vector g. We use the representations at the output of blocks 2, 3 and 4 as the local descriptors for estimating attention. For more details refer to the appendix.

We refer to the network *Net* with attention at the last level as *Net-att*, attention at the last two levels as *Net-att2* and attention at three levels as *Net-att3*. We specify *dp* for the use of dot-product for matching the global and local descriptors and *pc* for the use of parameterized compatibility. We specify *concat* for concatenation of descriptors from different levels for the final linear classification step. Alternatively, we specify *indep* for an independent prediction of probabilities at different levels using separate linear classifiers. The probabilities are later averaged to obtain a single score per class.

Task-specific processing: We generate the adversarial images using the fast gradient sign method of Goodfellow et al. (2014) and observe the network fooling behaviour at increasing L_∞ norms

of the perturbations. For cross-domain classification, we extract features at the input of the final fully-connected layer in each model and use these to train a linear SVM with $C = 1$ similar to the setup used by Zhou et al. (2016). We report the results of a 5-fold cross validation. At no point do we fine-tune the networks on the target dataset. For weakly supervised segmentation, we combine the attention maps from the last 2 levels using element-wise multiplication, take square-root of the result to re-interpret it as a probability distribution (in the limit of the two attention distributions approaching each other), and binarise the result using the Otsu binarization threshold.

Details of the dataset pre-processing, architecture modifications for ResNet and network training can be found in §A.

5 Results and Discussion

5.1 Image classification and fine-grained recognition

Model	Top-1 error with standard deviation. CIFAR-10	CIFAR-100
— Existing architectures —		
VGG (Simonyan & Zisserman, 2014)	7.77 (0.08)	30.62 (0.16)
VGG-GAP (Zhou et al., 2016)	9.87 (0.10)	31.77 (0.13)
RN-164 (He et al., 2015)	**6.03 (0.18)**	**25.34 (0.16)**
— Architectures with attention —		
(VGG-att)-dp	6.14 (0.06)	24.22 (0.08)
(VGG-att)-pc	5.67 (0.04)	23.70 (0.07)
(VGG-att2)-indep-dp	5.91 (0.05)	23.24 (0.07)
(VGG-att2)-indep-pc	5.36 (0.06)	24.00 (0.06)
(VGG-att2)-concat-pc	**5.23 (0.04)**	23.19 (0.04)
(VGG-att3)-concat-pc	6.34 (0.07)	**22.97 (0.04)**

Table 1: CIFARs: Top-1 classification errors.

Model	Top-1 error with standard deviation. CUB-200-2011	SVHN
— Existing architectures —		
VGG (Simonyan & Zisserman, 2014)	34.64 (0.26)	4.27 (0.04)
VGG-GAP (Zhou et al., 2016)	29.50* (–)	5.84 (0.09)
— Architectures with attention —		
(VGG-att2)-concat-pc	**26.80 (0.16)**	3.74 (0.05)
(VGG-att3)-concat-pc	26.95 (0.10)	**3.52 (0.04)**

Table 2: Fine grained recognition: Top-1 errors. {*} is taken from paper.

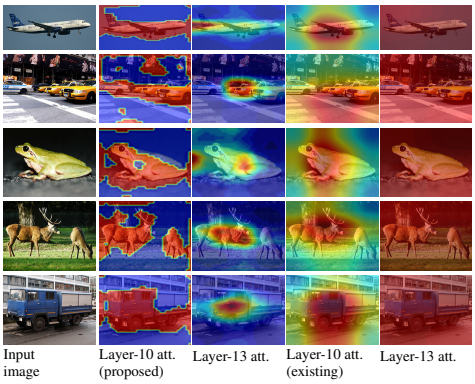

Figure 3: Attention maps from VGG-att2 model trained on low-res CIFAR-10 dataset focus sharply on the objects of interest in high-res ImageNet images for cifar categories; contrasted here with the activation-based attention maps of Zagoruyko & Komodakis (2016).

Figure 4: VGG-att2 trained on CUBS-200 for fine-grained bird recognition task: layer-10 learns to fixate on eye and beak regions, layer-13 on plumage and feet.

Incorporating attention in a standard VGG architecture provides noticeable performance improvement over baseline and other counterparts *e.g.*, VGG-GAP, ResNet(RN) as seen for the classification task on CIFAR datasets in Table 1. Specifically, *VGG-att2-concat-pc* model achieves a 2.5% and 7.4% improvement over baseline for CIFAR-10 and CIFAR-100. For the task of fine-grained recognition of CUB-200-2011 and SVHN categories the same model gives 7.8% and 0.5% improvement w.r.t. vanilla VGG, see Table 2. Thus, for the rest of the experiments *concat-pc* is the implicit design of attention unless specified otherwise. We visualise the attention maps learned by this network over the earlier layers in Fig. 3 . Observably, the attention mechanism enables the network to focus on the object of interest which suppressing the background regions. For the task of fine-grained recognition, the same layers learn specialized focus on different object parts as seen in Fig. 4. As per our intuition, adding attention at more than two levels starts to degrade performance. Note that due to the inclusion of additional layers, the number of parameters in the attention model is more

than in the baseline. However, training under the constraints of the attention mechanism allows the networks to generalise well on the test set.

When the same attention mechanism is introduced in ResNets (RN) we observe a marginal drop in performance, 0.9% on CIFAR-10 and 1.5% on CIFAR-100. We believe that the skip-connections in these models that combine the features from shallow network layers with those of deeper layers could be working in a manner similar to the proposed attention mechanism *i.e.*allowing the network a systematic focus on some local regions while downplaying others. However, unlike the skip-connection architecture, our method is able to learn explicit maps of network attention and exploit these for other tasks such as weak segmentation.

5.2 Robustness to adversarial attack

Figure 5: Adversarial versions of a sample input image for log-linearly increasing L_∞ norm from 1 to 16, estimated for VGG and VGG-att2 trained on CUBS-200.

L_∞ norm	Fooling rate	
	VGG	VGG-att2
1	58.58	53.78
2	78.27	76.00
4	89.99	89.90
8	94.89	95.27
16	96.46	97.60

Figure 6: Network fooling rate measured as a percentage change in the predicted class labels w.r.t those predicted for the unperturbed images.

From Fig. 6, the fooling rate of attention aware VGG is 5% less than the baseline VGG at an L_∞ noise norm of 1. As the norm increases, the fooling rate saturates for the two networks and the gap gradually decreases. Interestingly, when the noise begins to be perceptible (see col.5, Fig. 5), the fooling rate of VGG-att2 is around a percentage higher than that of VGG.

5.3 Cross-domain image classification

Model	Top-1 accuracies using models trained on CIFAR-10 / CIFAR-100.						
	STL-train	STL-test	Caltech-101	Caltech-256	Event-8	Action-40	Scene-67
— Existing archi. —							
VGG	45.34 / –	44.91 / –	35.97 / 54.20	13.16 / 25.57	53.62 / 57.05	13.85 / 17.58	11.02 / 16.73
VGG-GAP	43.24 / –	42.76 / –	41.68 / 62.61	16.30 / 31.39	58.83 / 68.11	16.73 / 24.50	12.85 / 23.04
RN-164	47.82 / –	47.02 / –	49.89 / 73.62	22.59 / 39.65	69.19 / 75.10	20.56 / 28.72	20.26 / 29.89
— Attention archi. —							
VGG-att3	**48.42** / –	**48.32** / –	58.34 / 75.39	29.99 / 44.14	77.06 / 82.08	26.75 / 30.96	26.86 / 33.72
RN-att2	46.36 / –	46.45 / –	**68.11 / 79.17**	**36.33 / 46.20**	**80.37 / 83.67**	**30.47 / 31.45**	**30.46 / 34.39**

Table 3: Cross-domain classification: Top-1 accuracies using models trained on CIFAR-10/100.

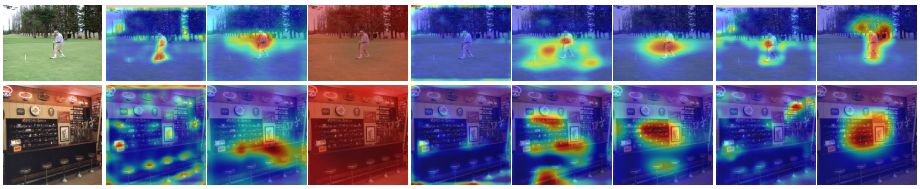

Figure 7: Row 1 : Event-8 (croquet), Row 2 : Scene-67 (bar). Attention maps for models trained on CIFAR-100 (c100) are more diverse than those from the models trained on CIFAR-10 (c10). Note the sharp attention maps in col.7 versus the uniform ones in col.4. Attention maps at lower levels appear to attend to part details (*e.g.*the stack of wine bottles in the bar (row 2)) and at a higher level on whole objects owing to a large effective receptive field.

CIFAR images cover a wider variety of natural object categories in comparison to SVHN and CUB, hence we use these to train different network architectures and use the networks as off-the-shelf feature extractors to evaluate their generalizability on new unseen datasets. As seen in Table 3, attention aware models improve over the baseline models by an average margin of 6% across all unseen datasets. We make two additional observations. Firstly, low resolution CIFAR images contain useful visual properties that are transferrable and useful in classifying high resolution images such as the 600×600 images of Event-8 dataset (Li & Fei-Fei, 2007). Secondly, training for diversity is better. CIFAR-10 and CIFAR-100 datasets contain same corpus of images organized differently into 10 and 100 categories respectively. From the accuracy pairs in Table 3, and the attention maps of Fig. 7, it appears that while learning to distinguish a larger set of categories the network is able to capture more nuanced features that transfer well to classifying new datasets. (Note that STL dataset shares the same category set as CIFAR-10. Hence, attention-mounted and baseline models are directly evaluated on STL without the use of an SVM.)

5.4 Weakly-supervised semantic segmentation

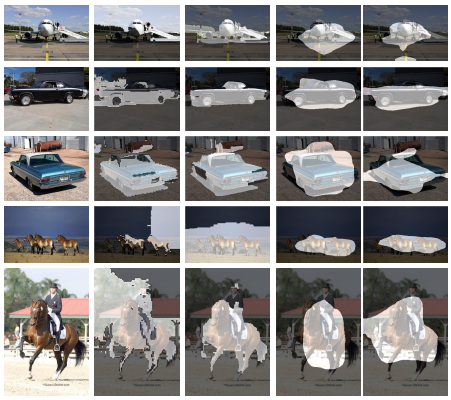

Figure 8: Weakly supervised segmentation by binarising attention maps.

Models	Airplane	Car	Horse
— Attention-based —			
VGG-GAP (CIFAR-10)	10.10	19.81	29.55
VGG-att2 (CIFAR-10)	**48.07**	61.19	**40.95**
VGG-att2 (CIFAR-100)	41.10	48.35	37.93
RN-att2 (CIFAR-10)	41.01	63.12	36.78
RN-att2 (CIFAR-100)	45.46	**65.05**	40.36
— Saliency-based —			
Jiang *et al.* (Jiang et al., 2013)	37.22	**55.22**	**47.02**
Zhang *et al.* (Zhang & Sclaroff, 2013)	**51.84**	46.61	39.52
— Top object proposal-based —			
MCG (Arbeláez et al., 2014)	**32.02**	**54.21**	**37.85**
— Joint segmentation-based —			
Joulin *et al.* (Joulin et al., 2010)	15.36	37.15	30.16
Object-discovery (Rubinstein et al., 2013)	55.81	64.42	51.65
Chen *et al.* (Chen et al., 2014)	54.62	**69.20**	44.46
Jain *et al.* (Dutt Jain & Grauman, 2016)	**58.65**	66.47	**53.57**

Figure 9: Jaccard scores (higher is better) for binarised attention maps from CIFAR-10/100 trained models tested on the Object Discovery dataset.

From Table 9, the proposed attention maps perform significantly better at weak segmentation than those obtained using the GAP approach of Zhou et al. (2016); the minimum improvement in terms of Jaccard index being 30%. We compare favourably to the top object-proposal based approach; outperforming for all three categories of airplane, car and horse by a minimum margin of 3%. Note that we do not compare with the state-of-the-art CNN-based object proposal approaches given that they are trained using additional bounding box annotations. We surpass the saliency-based approaches for segmentation of car category, but perform less well for the other two categories of airplane and horse. This could be due to the detailed structure and small size of instances of the latter two categories, see Fig. 8. We are not far behind the top performing methods that use joint inference on a group of test images for segmenting out the common object category.

6 Conclusion

We propose a trainable attention module for generating probabilistic landscapes that highlight where and in what proportion a network attends to different regions of the input image for the task of classification. We demonstrate that the method, when deployed at multiple levels within a network, affords significant performance gains in classification of seen and unseen categories by focusing on the object of interest. We also show that the attention landscapes can facilitate weakly supervised segmentation of the predominant object. Further, the proposed attention scheme is amenable to popular post-processing techniques such as conditional random fields for refining the segmentation masks, and has shown promise in learning robustness to certain kinds of adversarial attacks.

REFERENCES

Pablo Arbeláez, Jordi Pont-Tuset, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 328–335, 2014.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Chunshui Cao, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, Deva Ramanan, and Thomas S. Huang. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *ICCV*, 2015.

Xinlei Chen, Abhinav Shrivastava, and Abhinav Gupta. Enriching visual knowledge bases via object discovery and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2027–2034, 2014.

Adam Coates, Honglak Lee, and Andrew Y Ng. An analysis of single-layer networks in unsupervised feature learning [http://cs.stanford.edu/ acoates/stl10]. *Ann Arbor*, pp. 2, 2010.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *Icml*, volume 32, pp. 647–655, 2014.

Suyog Dutt Jain and Kristen Grauman. Active image segmentation propagation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2864–2873, 2016.

M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, January 2015.

Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006.

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *CoRR*, abs/1412.6572, 2014. URL `http://arxiv.org/abs/1412.6572`.

Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pp. 2017–2025, 2015.

Bowen Jiang, Lihe Zhang, Huchuan Lu, Chuan Yang, and Ming-Hsuan Yang. Saliency detection via absorbing markov chain. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1665–1672, 2013.

Armand Joulin, Francis Bach, and Jean Ponce. Discriminative clustering for image co-segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 1943–1950. IEEE, 2010.

Armand Joulin, Francis Bach, and Jean Ponce. Multi-class cosegmentation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 542–549. IEEE, 2012.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009.

Li-Jia Li and Li Fei-Fei. What, where and who? classifying events by scene and object recognition. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pp. 1–8. IEEE, 2007.

Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *Advances in Neural Information Processing Systems*, pp. 2204–2212, 2014.

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.

Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 413–420. IEEE, 2009.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *arXiv preprint arXiv:1506.02640*, 2015.

Michael Rubinstein, Armand Joulin, Johannes Kopf, and Ce Liu. Unsupervised joint object discovery and segmentation in internet images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1939–1946, 2013.

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

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, California Institute of Technology, 2011.

Beilun Wang, Ji Gao, and Yanjun Qi. A theoretical framework for robustness of (deep) classifiers under adversarial noise. *arXiv preprint arXiv:1612.00334*, 2016.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning (ICML) 2 (3), 5*, 2015.

Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lai Lin, Leonidas Guibas, and Li Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 1331–1338. IEEE, 2011.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *CoRR*, abs/1612.03928, 2016.

Jianming Zhang and Stan Sclaroff. Saliency detection: A boolean map approach. In *Proceedings of the IEEE international conference on computer vision*, pp. 153–160, 2013.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2921–2929, 2016.

A APPENDICES

A.1 DATASETS

We evaluate the proposed attention models on CIFAR-10 (Krizhevsky & Hinton, 2009), CIFAR-100 (Krizhevsky & Hinton, 2009), SVHN (Netzer et al., 2011) and CUB-200-2011 (Wah et al., 2011) for the task of classification. We use the attention incorporating VGG model trained on CUBS-200-2011 for investigating robustness to adversarial attacks. For cross-domain classification, we test on 6 standard benchmarks including STL (Coates et al., 2010), Caltech-256 (Griffin et al., 2007) and Action-40 (Yao et al., 2011). We use the Object Discovery dataset (Rubinstein et al., 2013) for evaluating weakly supervised segmentation performance. A detailed summary of the these datasets can be found in 4.

For all datasets except SVHN and CUB, we perform mean and standard-deviation normalization as well as color normalization. For CUB-200-2011, the images are cropped using ground truth bounding box annotations and resized. For cross-domain image classification we downsample the input images to avoid memory overhead.

Dataset	Size (total/train/test/extra)	Number of classes / Type	Resolution	Tasks
CIFAR-10 (Krizhevsky & Hinton, 2009)	60,000 / 50,000 / 10,000 / –	10 / natural images	32x32	C, C-c, S
CIFAR-100 (Krizhevsky & Hinton, 2009)	60,000 / 50,000 / 10,000 / –	100 / natural images	32x32	C, C-c, S
SVHN (Netzer et al., 2011)	– / 73,257 / 26,032 / 531,131	10 / digits	32x32	C
CUB-200-2011 (Wah et al., 2011)	– / 5,994 / 5,794 / –	200 / bird images	80x80	C
STL (Coates et al., 2010)	– / 5,000 / 8,000 / –	10 / ImageNet images	96x96	C-c
Caltech-101 (Fei-Fei et al., 2006)	8677 / – / – / –	101 / natural images	$\sim$300x200	C-c
Caltech-256 (Griffin et al., 2007)	29,780/ – / – / –	256 / natural images	$\sim$300x200	C-c
Event-8 (Li & Fei-Fei, 2007)	1574 / – / – / –	8 / in/out door sports	$>$600x600	C-c
Action-40 (Yao et al., 2011)	9532/ – / – / –	40 / natural images	$\sim$400x300	C-c
Scene-67 (Quattoni & Torralba, 2009)	15613 / – / – / –	67 / indoor scenes	$\sim$500x300	C-c
Object Discovery (Rubinstein et al., 2013)	300/ – / 300 / –	3 / synthetic and natural images	$\sim$340x240	C-c, S

Table 4: Summary of datasets used for experiments across different tasks (C: classification, C-c: classification cross-domain, S: segmentation).The natural images span from those of objects in plain background to cluttered indoor and outdoor scenes. The objects vary from simple digits to humans involved in complex activities.

A.2 NETWORK ARCHITECTURES

ResNet: Our baseline ResNet implementation contains 4 distinct levels that map the RGB input into a 256-dimensional space through 16, 64 and 128 dimensional embedding spaces respectively. Each level excepting the first, which contains 2 convolutional layers separated by a non-linearity, contains n-residual blocks and each residual block contains a maximum of 3 convolutional layers interleaved by non-linearities. This yields a network definition of $9n + 2$ parameterized layers (He et al., 2015). We work with an n of 18 for a 164-layered network.

Attention in ResNet: We replace the spatial average pooling layer after the final level 4 by convolutional and max-pooling steps which gives us our global feature vector g. We refer to the network with attention on the output of the last level as RN-att and attention at the output of last two levels as RN-att2. Following the results of VGG network, we train these attention units in the *concat-pc* framework.

A.3 TRAINING ROUTINES

VGG networks for CIFAR-10, CIFAR-100 and SVHN are trained from scratch. We use a stochastic gradient descent (SGD) optimizer with a batch size of 128, initial learning rate of 1, learning rate decay of 10^{-7}, weight decay of 5×10^{-4}, and momentum of 0.9. The learning rate is scaled by 0.5 every 25 epochs and we train over 300 epochs for convergence. For CUB, since the training data is limited, we initialise the model with the weights learned for CIFAR-100. We use the transfer-learning training schedule inspired by Redmon et al. (2015). Thus the training starts at a learning rate of 0.1 for first 30 epochs, is multiplied by 2 twice over the next 60 epochs, and then scaled by 0.5 every 30 epochs for the next 200 epochs.

For ResNet, the networks are trained using an SGD optimizer with a batch size of 64, initial learning rate of 0.1, weight decay of 5×10^{-4}, and a momentum of 0.9. The learning rate is multiplied by 0.2 after 60, 120 and 160 epochs. The network is trained for 200 epochs until convergence. We train the models for CIFAR-10 and CIFAR-100 from scratch.

All models are implemented in Torch and trained with an NVIDIA Titan-X GPU. Training takes around one to two days depending on the model and datasets.