On the Relationship between Visual Attributes and Convolutional Networks

Victor Escorcia[1,2], Juan Carlos Niebles[2], Bernard Ghanem[1]
[1]King Abdullah University of Science and Technology (KAUST), Saudi Arabia
[2]Universidad del Norte, Colombia

Abstract

One of the cornerstone principles of deep models is their abstraction capacity, i.e. their ability to learn abstract concepts from 'simpler' ones. Through extensive experiments, we characterize the nature of the relationship between abstract concepts (specifically objects in images) learned by popular and high performing convolutional networks (conv-nets) and established mid-level representations used in computer vision (specifically semantic visual attributes). We focus on attributes due to their impact on several applications, such as object description, retrieval and mining, and active (and zero-shot) learning. Among the findings we uncover, we show empirical evidence of the existence of Attribute Centric Nodes (ACNs) within a conv-net, which is trained to recognize objects (not attributes) in images. These special conv-net nodes (1) collectively encode information pertinent to visual attribute representation and discrimination, (2) are unevenly and sparsely distribution across all layers of the conv-net, and (3) play an important role in conv-net based object recognition.

1. Introduction

The seminal work of Krizhevsky *et al*. [9] that trained a large conv-net for image-level object recognition on the ImageNet challenge is considered a major stepping stone for subsequent work in conv-net based visual recognition. Such a network is able to automatically learn a hierarchy of non-linear features that richly describe image content as well as discriminate between object classes. Indeed, the excellent performance of such a network on this large-scale recognition task raises the question of whether intuitive and hand-crafted features such as SIFT and HOG can be replaced by conv-net features. In fact, recent work [2, 18] has shown that features extracted from a conv-net trained on ImageNet are general purpose (or black-box) enough to achieve state-of-the-art results in various other recognition tasks, including scene, fine-grained, and even action recognition. However, unlike hand-crafted features, those learned by a conv-net are usually not visually intuitive and straightforward to interpret. Despite their excellent recognition performance, understanding and interpreting the inner workings of conv-nets remains mostly elusive to the community. It is this lack of deep understanding that is currently motivating researchers to *look under the hood* and comprehend how and why these deep networks work so well in practice.

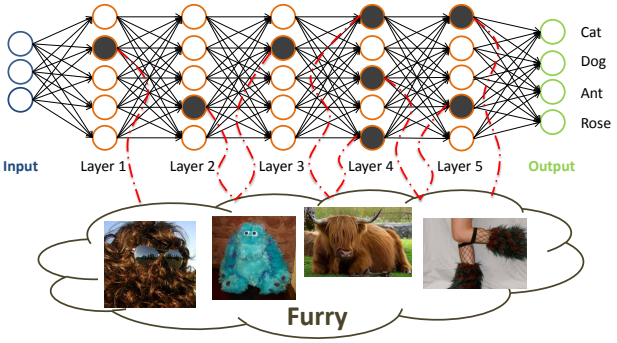

Figure 1. Given the versatility and excellent performance of conv-nets on various visual recognition tasks, it is plausible that mid-level representations such as visual attributes are encoded by the network's activations. The location and sparsity of this encoding, as well as, the general properties of the relationship between attributes and pre-trained conv-nets are the focus of this work. (All the figures are best viewed in color)

In very recent work [1, 23], the activations of a conv-net are put in the lime light and many interesting observations are made. For example, using a deconv-net, these activations can be visually analyzed now and different parts of the conv-net can be probed to measure their impact on overall recognition. Analysing the properties of these activations has shed light on better and more efficient strategies to pre-train, fine-tune, and design a conv-net. Inspired by these observations, this paper takes another step in a similar direction, namely understanding how the inner workings of a conv-net that is trained for a high-level recognition task (object recognition) relate to intuitive and conventional mid-level representations in computer vision literature.

Despite the insights of recent work, it is still unclear how visual content is represented within the activations of a conv-net. Simply put, a conv-net trained to classify objects in images can be viewed as a deep learning machine that

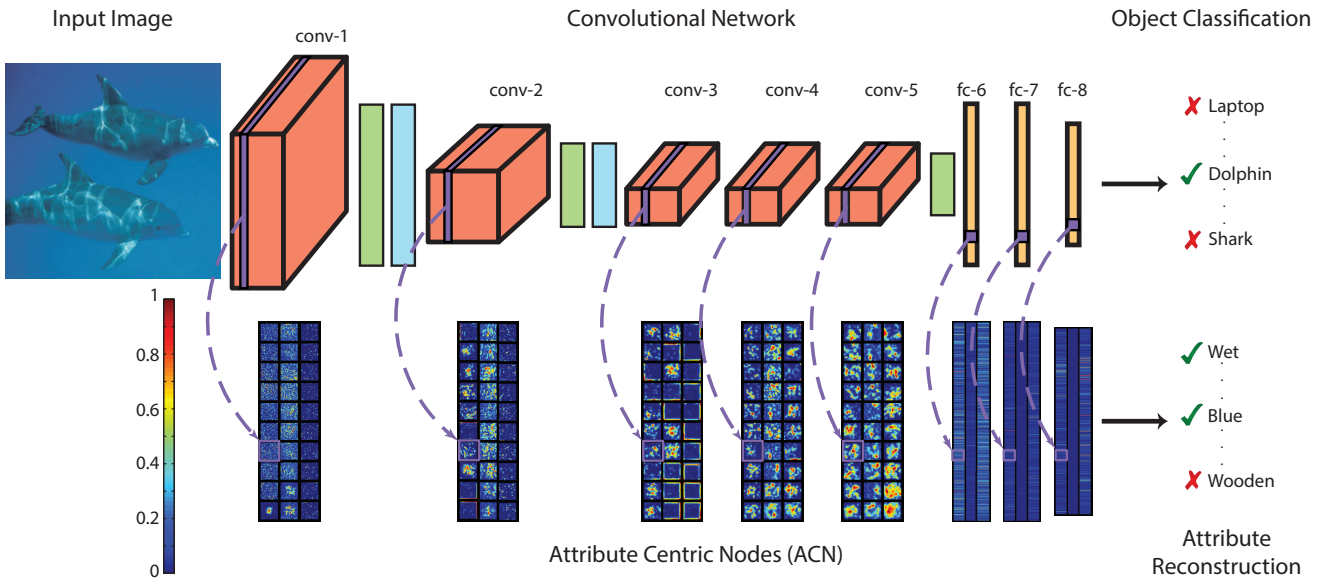

Figure 2. In this work, we empirically show that a sparse number of nodes in a conv-net, trained to recognize objects (e.g. 'dolphin'), inherently encode information about semantic visual attributes (e.g. 'wet'). We call these activation locations Attribute Centric Nodes (ACNs), which can be discovered by solving an ℓ_1 minimization problem (μ-LASSO).

finds the appropriate mapping between input (raw pixel intensities) and output (object labels) layers. It is conceivable to ask here whether this mapping makes use of a mid-level representation for objects, similar in spirit to how the human visual system functions. In fact, the empirically validated and general purposefulness of conv-net activations across different visual recognition tasks [18, 2] suggest that such a shared mid-level representation of the visual world is being automatically learned. More importantly, it is worthwhile to investigate whether this learned mid-level representation is related to (if at all) intuitive mid-level representations (e.g. parts[1] [4], mid-level patches [20], and visual attributes [3]) innovated by the community before conv-nets were popularized. Since addressing these queries in their entirety is beyond the scope of this paper, we focus on studying the relationship between a conv-net trained to recognize objects in images and object-level visual attributes. This can help us realize, for example, whether a conv-net trained to recognize a 'dog' inherently learns what 'fluffy' means *without* prior knowledge of the attribute.

Main Findings: In this work, we hypothesize that a sparse number of nodes in a deep conv-net trained for image-level object recognition on ImageNet can reliably predict absolute visual attributes [3]. Through rigorous experimentation, we uncover the following aspects of the relationship between such a conv-net and visual attributes.

(1) Visual attributes can be predicted reliably using a sparse number of nodes from the conv-net. This suggests that the conv-net can *indirectly* learn attribute concepts, even though it is trained to recognize objects, as depicted in Figure 1.

(2) Nodes in the conv-net that are used to represent attributes are called Attribute Centric Nodes (ACNs) which are illustrated in Figure 1. The support of ACNs in the network is sparse and unevenly distributed among the different layers (convolutional and fully-connected). On average, these ACNs are concentrated in the top layers of the network; however, their exact locations are attribute dependent. Also, attributes that co-occur in images (e.g. 'furry' and 'black'/'brown') share ACNs.

(3) The recognition accuracy of ImageNet objects is significantly reduced when ACNs of all attributes are ablated from the conv-net, significantly more so, than when an equal number of randomly sampled nodes are ablated. This suggests that conv-nets actually make use of learned attribute representations (through ACNs) to recognize objects in images. Interestingly, ablating ACNs corresponding to a specific set of attributes (e.g. 'furry', 'black', and 'brown') has the most effect on object classes that are described by these attributes (e.g. 'retriever' and 'gazelle') and the least effect on classes that are not (e.g. 'bathtub' and 'chain').

Related work

Visual Attributes: They identify mid-level concepts useful for describing semantic entities like objects, scenes, activities, etc. The seminal works of Farhadi *et al*. [3] and Lampert *et al*. [10] show the advantages of expanding the typical recognition problem of entities to recognizing visual attributes in images. Attributes have been shown to be useful in describing known and unknown entities, detect-

[1]The relationship between conv-nets and the deformable parts model (DPM) has been recently studied in [6].

ing atypical entities, learning models of unseen object categories from textual descriptions (zero-shot learning) and improving the recognition performance of entities (e.g. objects and scenes) in certain scenarios [3, 22]. Unlike these binary attributes, other work focuses on the merits of relative attributes [13], which mine pairwise relationships between attributes of different images. In this paper, we focus on object-level binary attributes and study their relationship with activations of a conv-net trained to recognize objects.

Understanding Conv-Nets: A conv-net is a special type of multi-layer neural network that uses a convolutional operator to combine activations from previous layers. Since the topic of deep learning using conv-nets is rich in the literature, we refer the reader to the seminal work of [11] and a recent survey [17] for a more detailed discussion. The most successful conv-net architectures are trained with backpropagation and used recently developed regularization techniques [9, 23]. Although these models have been shown to learn a richer and more discriminative feature mapping than hand-crafted conventional features across various vision tasks [5, 2, 8], it remains unclear what types of visual features are learned in the pool of their hidden layers. This lack of understanding has encouraged the community to go beyond a 'black-box' view of conv-nets and seek deeper insight into their inner workings. In this spirit, Simonyan *et al*. [19] address the problem of finding images that activate specific nodes in the conv-net. This network probing scheme is useful for visualizing and diagnosing different parts of the conv-net's layers. Motivated by the same goal, recent work in [23] proposes a new scheme to visualize activations based on deconv-nets. Agrawal *et al*. [1] study several properties of conv-net activations to make conv-net design and training more intuitive and accessible for the community. One interesting finding is the existence of "grandmother cells" for a limited set of object classes.

Inspired by previous work that focuses on visualizing and probing a conv-net for a particular recognition task, we focus on studying the aspects of the relationship between a pre-trained conv-net and one popular mid-level representation (visual attributes). Specifically, we investigate the attribute prediction power of this conv-net, the localized encoding of attributes throughout the network, and the impact of encoded attribute concepts on object recognition. We hope our work encourages researchers to investigate this relationship for other forms of mid-level representations.

2. Approach

In this paper, we consider a conv-net that is pre-trained to recognize objects in images. This training is done on the ImageNet2012 challenge with 1000 object classes using a popularized conv-net architecture, whose specific details are discussed later in Section 3.1. As depicted in the top part of Figure 2, an input image can be fed forward through this conv-net to produce a soft-max prediction vector of the same size as the output label space (i.e. $\mathbb{R}^{1000}$ in our case). This manifestation of the conv-net comprises 5 convolutional layers denoted *conv1-conv5*, two fully connected (multi-layer perceptrons) layers denoted *fc6-fc7*, and an output layer denoted *fc8*. The activations of the conv-net at each of its layers can be viewed as a nonlinear mapping of the image into a higher-dimensional feature space. In fact, features from layers *fc6-fc8* have been used in previous work to train a discriminative model for various other visual tasks [18].

Since this work focuses on uncovering properties that describe the relationship between the inner workings of the conv-net and binary visual attributes, we model an input image using *all* m activations in the network. In our experiments, $m \approx 660$K. As such, the i^{th} image in a dataset is represented as a high-dimensional vector $\mathbf{x}_i \in \mathbb{R}^m$. Note that there is a one-to-one correspondence between each of the m elements in $\mathbf{x}_i$ and the location of the node from which the activation originated. Furthermore, the visual attributes describing this image are organized in a binary vector $\mathbf{l} \in \{0,1\}^d$, where d is the total number of attributes under consideration. In our experiments, $d = 25$. Given a set of N images $\{\mathbf{x}_i, \mathbf{l}_i\}_{i=1}^N$, we hypothesize that the j^{th} visual attribute of an image can be predicted by a simple linear combination $\mathbf{w}_j \in \mathbb{R}^m$ of its conv-net activations[2]. To prevent overfitting (since $N \ll m$) and to impose possible priors on individual and groups of attributes, we use a regularization function $g(\mathbf{W})$, where $\mathbf{W} = [\mathbf{w}_1 \cdots \mathbf{w}_d] \in \mathbb{R}^{m \times d}$. Learning these parameters can be formulated as solving the optimization problem in Eq (1), where $f(a,b)$ defines a loss function between a and b.

$$\mathbf{W}^* = \underset{\mathbf{W}}{\operatorname{argmin}} \sum_{i=1}^{N} \sum_{j=1}^{d} f(\mathbf{w}_j^T \mathbf{x}_i, \mathbf{l}_{ij}) + g(\mathbf{W}) \quad (1)$$

This conventional formulation for learning parameters is general, since the loss function f and regularizer g can take on various forms. To simplify the optimization at such a large scale, we use an ℓ_2 squared loss: $f(a,b) = (a-b)^2$ in this paper. Other losses (e.g. hinge loss) can be used without loss of generality. As for the regularizer, we model $g(\mathbf{W}) = \sum_{j=1}^{d} h_{\mu_j}(\mathbf{w}_j)$, where $h_\mu(\mathbf{y}) = \mathbb{1}_{\{\|\mathbf{y}\|_1 \leq \mu\}}$ is the indicator function for a hard ℓ_1 constraint. We use this regularization scheme to impose sparsity on $\mathbf{W}$, since we hypothesize that only a small number of nodes (called Attribute Centric Nodes or ACNs) in the conv-net are needed to reliably predict each attribute. The threshold μ_j directly controls the magnitude of this sparsity and it can be fine-tuned through cross-validation. Other regularizers can be used, each imposing certain priors. For example, a group sparsity term can be added to each $\mathbf{w}_j$ to force ACNs to exist in a small number of node groups, where a node group

[2] A bias term can be easily incorporated by appending 1 to each $\mathbf{x}_i$.

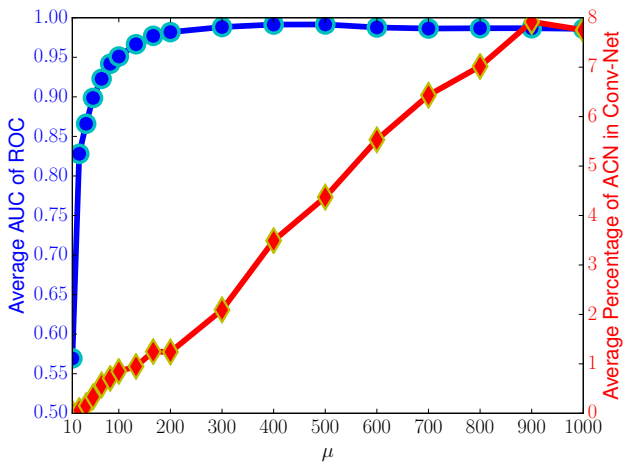

Figure 3. Plots the AUC-ROC (and percentage of ACNs in the conv-net) for attribute reconstruction on the ImageNet-Attribute dataset for various values of μ. Clearly, only a small percentage of ACNs is required to produce near perfect reconstruction. At $\mu = 200$ (1.2% of the conv-net nodes), the AUC-ROC score stabilizes, which is evidence that ACNs are truly sparse in the conv-net.

could be a complete layer in the conv-net or parts of it. Also, as we will see from our experimental results, a spectral regularizer e.g. a quadratic graph Laplacian term of the form $trace(\mathbf{WAW}^\top)$, can be incorporated to mine correlations *between* attribute parameters. Putting all these terms together, we resort to solving Eq (2).

$$\mathbf{W}^* = \underset{\mathbf{W}}{\text{argmin}} \ \|\mathbf{X}^\top\mathbf{W} - \mathbf{L}^\top\|_F^2 \qquad (2)$$
$$\text{subject to: } \|\mathbf{w}_j\|_1 \leq \mu_j \ \ \forall j = 1, \cdots, d$$

Here, we take $\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_N] \in \mathbb{R}^{m\times N}$ and $\mathbf{L} = [\mathbf{l}_1 \cdots \mathbf{l}_N] \in \mathbb{R}^{d\times N}$. Eq (2) is the matrix form for the popular μ-LASSO problem that can be efficiently solved (especially at large-scales) using the spectral gradient-projection (SPG) method [21]. Following common practice, we pre-normalize the rows of $\mathbf{X}$ to unit norm. To describe the size of this problem, the activation matrix $\mathbf{X}$ in our experiments requires more than 40GB to load into RAM memory. After solving Eq (2), the non-zero elements (up to a small threshold) of $\mathbf{w}_j^*$ (its sparse support) localize the ACNs of attribute j in the conv-net (refer to Figure 2). In what follows, we perform extensive experiments to uncover the properties of $\mathbf{W}^*$, specifically in regards to its ability to predict attributes in images, the distribution of ACNs across conv-net layers, and the effect of ACNs on object recognition.

3. Experiments and Discussion

3.1. Experimental Setup

We conduct three different experiments in order to validate whether and where information about visual attributes

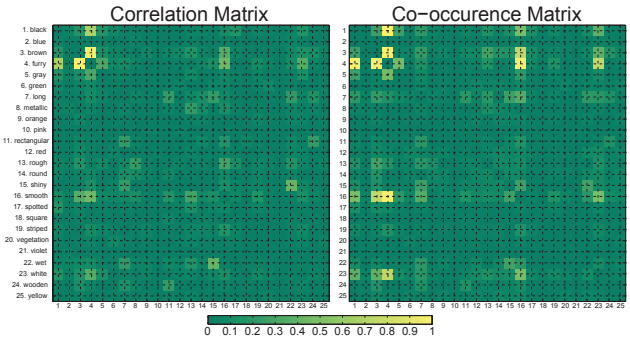

Figure 4. Compares the pairwise co-occurrence of attributes to the correlation of ACN sparse solutions on the ImageNet-Attribute dataset. Correlation values between sparse solutions tend to be proportional to the co-occurrence of their corresponding attributes, i.e. higher correlations tend to imply more frequent co-occurrences and vice versa.

is *inherently* encoded in a conv-net trained to recognize objects and the relationship of this encoding (through ACNs) with the task of object recognition.

Implementations details: To test our hypotheses, we need a large-scale dataset that contains images with both object and attribute labels. We choose the ImageNet dataset for this purpose. A subset of ImageNet (denoted the ImageNet-Attribute dataset [16]) comprises 9600 images labeled with 25 binary attributes.

As for the object recognition conv-net, we use the popular *Alex-net* architecture depicted in Figure 2. To enable the reproducibility of our results, we use the publicly available Caffe model [7]. This network has 5 convolutional layers followed by 3 fully-connected layers. Using a shorthand notation the full architecture is $C(96, 11, 4) - N - P - C(256, 5, 1) - N - P - C(384, 3, 1) - C(384, 3, 1) - C(256, 3, 1) - P - FC(4096) - FC(4096) - FC(1000)$ where $C(d, f, s)$ indicates a layer with d filters of $d \times d$ size applied with a stride s. $FC(n)$ is a fully-connected layer with n nodes. All pooling layers P use a kernel of 3×3 with a stride of 2 pixels and all normalization layers N are defined according to [9]. Except for the last fully-connected layer, all convolutional and fully-connected layers use the rectified linear unit (ReLu) as the non-linear activation function. Each image is rescaled to 256×256 pixels, from which multi-scaled crops of 227×227 pixels are taken. Then, we normalize each image to zero mean. More details of this network are included in the **supplementary material**. This conv-net is trained on the ILSVRC-12 dataset, which is a subset of ImageNet comprising 1.2 million images and 1000 object classes [15]. Then, we feed the images from the ImageNet-Attribute and ILSVRC-12 datasets through the trained conv-net, while retaining all the activations (after ReLu) of all the convolutional and fully connected layers in the network ($m \approx 660K$).

Solving Eq (2) using SPG on a 3GHz 96GB RAM workstation running MATLAB takes 45 mins (on average) to

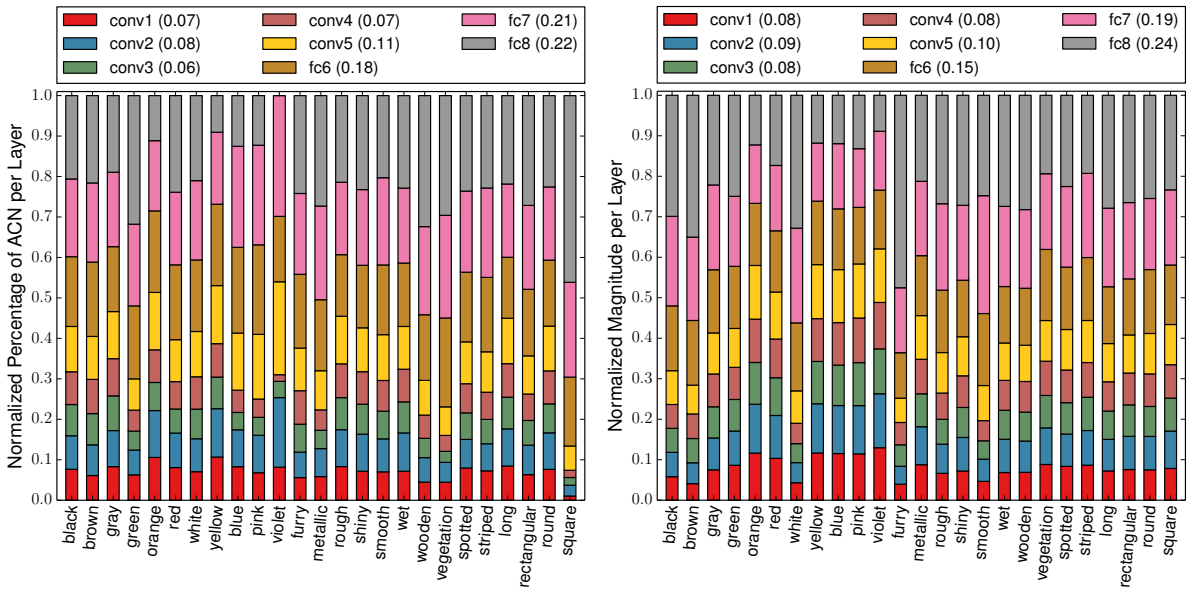

Figure 5. Shows contributions of each conv-net layer to attribute reconstruction. (*left*) For each attribute, we plot the normalized percentage of ACNs in each layer, scaled inversely proportional to its number of nodes. The average percentage of each layer across all attributes is in parentheses in the legend. (*right*) For each attribute, we plot the normalized ℓ_1 magnitude of each layer, scaled inversely proportional to its number of nodes. On average, the top layers of the conv-net tend to contribute more than the bottom ones.

converge using a reasonable stopping criterion. Of course, the larger μ is, the longer the runtime.

3.2. Does a conv-net *indirectly* learn attributes?

Attribute Reconstruction: We are interested in testing our hypothesis on the existence of a sparse number of ACNs that encode semantic attribute information. To do this, we solve Eq (2) using all the activations of the pre-trained conv-net for *all* the images in the ImageNet-Attribute dataset (with $\mu_j = \mu \ \forall j$), across a range of μ values. In this case, $N = 9600$, $d = 25$, and $m \approx 660$K. To measure how well the attributes are being reconstructed (i.e. how well ACNs encode attribute information), we compute the average area under the ROC curve (AUC-ROC) of all 25 attributes, for each value of μ. We report these results in Figure 3, which also plots the average percentage of ACNs selected within the conv-net. It is obvious that only a small percentage of the nodes in the conv-net (ACNs) is required to reliably encode attributes, at a high AUC-ROC value. Interestingly, even at $\mu = 10$ (i.e. using 3K nodes out of the total 660K), the AUC-ROC is still quite high at 51% (for all 25 attributes). A relatively small increase in μ (i.e. in the number of ACNs) quickly saturates the AUC-ROC score. These results provide evidence that a conv-net trained to recognize objects can *indirectly* learn a mid-level representation (visual attributes in this case).

Attribute Co-Occurrence: A single image can be described by more than one attribute. A pair of attributes are said to co-occur in an image if they both manifest in it. We compute the frequency of co-occurrence between all pairs of attributes on the ImageNet-Attribute dataset in the form of a normalized co-occurrence matrix (see Figure 4 (*right*)). For comparison, we also compute the correlation matrix $\mathbf{G} = \mathbf{W}^{*\top}\mathbf{W}^{*}$, which measures the linear correlation between pairs $(\mathbf{w}_i^*, \mathbf{w}_j^*)$ of sparse solutions from Eq (2) using $\mu = 500$ (see Figure 4 (*left*)). We observe two interesting properties. **(a)** The results show that attributes that frequently co-occur have similar ACN supports (since their sparse solutions have relatively high correlations), while those that do not co-occur much tend to have disjoint supports. For example, 'furry' co-occurs with 'black' & 'brown' *and* their corresponding sparse solutions have a consistently high correlation. The opposite relationship arises with 'vegetation' and 'violet' for example. One might argue that the very high-dimensionality of each $\mathbf{w}_i^*$ is the reason why such low correlations exist for most of the attribute solutions. However, even based on this argument, the presence of high correlation values that coincide with high co-occurrence values *cannot* be coincidental. **(b)** This finding hints at the possibility that the conv-net not only encodes attribute information, but it does so *efficiently* by encoding groups of co-occuring attributes together. Here, we note that this pairwise relationship arises even though the regularizer in Eq (2) does not enforce it. This motivates the use of other regularizer forms for $g(.)$ in Eq (2), which can embed structural priors, such as co-occurrence.

ACN Localization: After showing that a sparse number of ACNs exist in the conv-net, we aim to localize the layers in

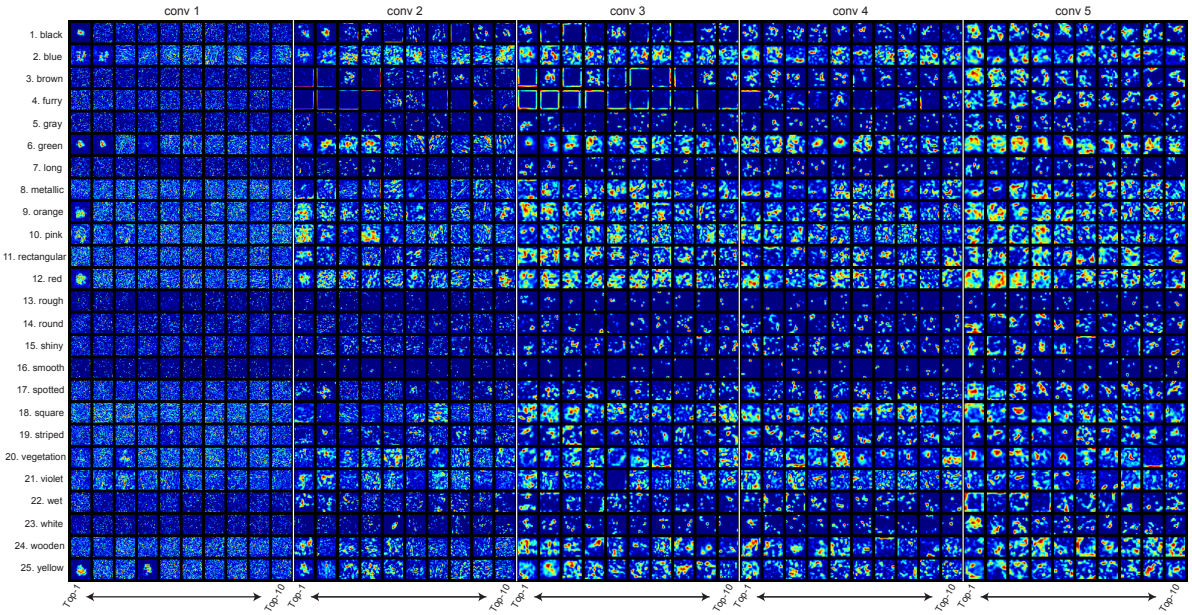

Figure 6. Visualizes the portions of the sparse ACN solution of Eq (2) across the different filters of each convolutional layer and each attribute. We use $\mu = 400$ here. Only the top-10 portions (in terms of ℓ_1 norm) of each layer are shown. Interestingly, ACNs in lower convolutional layers tend to be spatially structure-less, while they become more centralized and spatially contiguous in higher layers.

which they occur. To do this, we compute the normalized contribution of each layer to the overall sparse solution of each attribute. To distinguish the existence of an ACN from its 'magnitude' or importance (for each attribute), we define two measures for each layer: (i) the number of ACNs in the layer, and (ii) the ℓ_1 norm of the part of the sparse solution it corresponds to. Since the number of nodes varies across layers, we weigh both measures with a value that is inversely proportional to the total number of nodes in the layer. Then, for each attribute, we normalize the weighted measures of all layers to sum to one. Both normalized measures are shown in Figure 5 for all 25 attributes. In general, we find that more than 60% of the ACN count and magnitude contribute to the top 3 layers of the conv-net. This validates the results of some previous works [2, 18] that use activations from these top layers in different recognition tasks, including attribute detection. However, it is important to mention that the lower convolutional layers also play an important role in representing attributes, especially colors such as 'blue' or 'orange'. Interestingly, ACNs of texture attributes such as 'furry', 'metallic', and 'wooden' tend to be found in the higher layers.

ACN Visualization in Conv Layers: Since the activations of the convolutional layers encode spatial information directly, we visualize the absolute value of portions of the sparse solution (of each attribute) corresponding to the different filters in these layers. In each layer, we reshape these portions into 2D images of the same size as the filter activations and rank them in descending order according to their ℓ_1 norm. Figure 6 shows the top-10 filter portions per convolutional layer for each attribute. The sparsity of our solution (cold colors) and the distribution of the ACNs (hot colors) is evident. We find that the results in the lower layers (conv1 and sometimes conv2) are fragmented and have no clear structure, in general. Also, they tend to be more localized and spatially contiguous farther up in the network. However, this property is attribute specific. For example, attributes like 'black' and 'green', have interesting structures (spatially centralized and large magnitude) in the lower layers. This indicates that these attributes tend to be consistently spatially localized in images with 'black' and 'green' labels.

3.3. Can ACN activations predict attributes?

Previously, we showed that a sparse number of ACNs do preside in the conv-net in an unevenly distributed fashion. Now, we focus on studying the generalization performance of these ACNs in predicting visual attributes in unseen images. This task of attribute detection/recognition has been addressed in previous work [2, 5, 18], which use conv-net activations in the top-3 layers *only*. In this experiment, we show that activations at ACNs are not only reliably discriminative of attributes but that their discriminative power improves when *all* conv-net layers are considered.

To do this, we follow the same evaluation protocol suggested by Russakovsky *et al*. [16] on the ImageNet-Attribute dataset. We perform 5-fold cross-validation, where for each split of the data 3 folds are for training, 1 for validation (tuning μ_j for each attribute), and 1 for testing. We run 15 independent splits in this manner and report

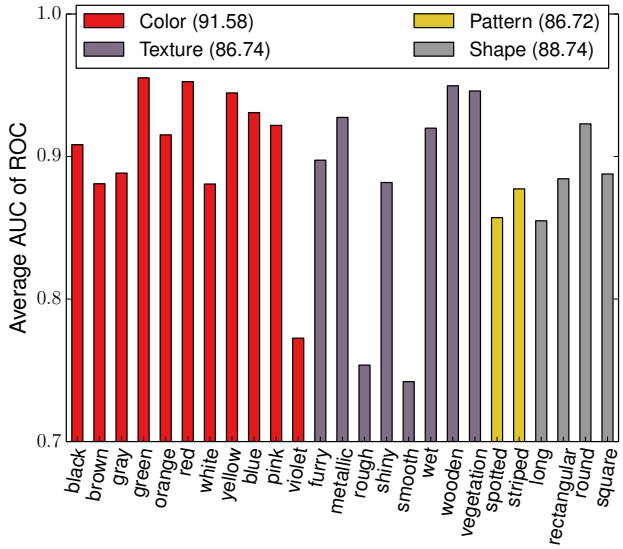

Figure 7. Shows attribute prediction performance measured in terms of the mean area under ROC curve. Attributes are grouped into 'color', 'pattern','texture', and 'shape' as suggested in [16]. The average performance of each group for the 20 attributes used by [16] is reported in parentheses.

Table 1. Summary of attribute recognition results on the ImageNet-Attribute dataset. We report the average area under the ROC curve for two setups: (*top*) 20 attributes (organized into 4 groups) as in [16] and (*bottom*) all 25 attributes (also in 4 groups). These results confirm the discriminative power and versatility of ACN activations for attribute recognition/prediction.

Attribute Group	[16]	**Our**
Color (8 attr)	87.5%	91.6%
Texture (7 attr)	77.5%	86.7%
Pattern (2 attr)	63.4%	86.7%
Shape (3 attr)	83.6%	88.7%
Overall (20 attr)	80.8%	**89.0%**

Attribute Group	**Our**	SVM FC-6	SVM FC-7	SVM FC-8
Color (11 attr)	90.5%	88.1%	89.6%	87.5%
Texture (8 attr)	87.7%	84.3%	83.9%	85.5%
Pattern (2 attr)	86.7%	87.6%	86.9%	85.9%
Shape (4 attr)	88.8%	90.4%	90.9%	89.0%
Overall (25 attr)	**89.0%**	87.2%	87.7%	86.9%

the mean area under the ROC curve for all splits. Figure 7 summarizes our results. Clearly, the proposed sparse representation of ACN activations generalizes well for attribute recognition. In Table 1, we compare our performance with the work in [16], which uses a non-linear SVM with multiple hand-crafted features (color, SIFT and shape context) to predict only 20 of the 25 attributes. Similar to [16], we organize the attributes into four attribute groups to simplify visualization. We record an average improvement of 9.8% across all 20 attributes. We also compare our results on the 25 attribute case with that of a linear SVM trained *only* on activations of the top 3-layers of the conv-net. We conclude that activations in the convolutional layers help improve recognition performance slightly ($1.3 - 2.1\%$).

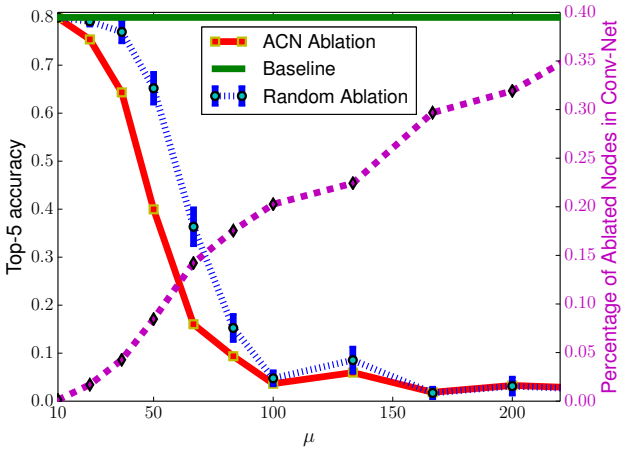

Figure 8. Plots top-5 accuracy on the ILSVRC-2012 validation set of the original *Alexnet* model (*green*) and its two surgically damaged variants. One variant (*red*) ablates the ACNs of all 25 attributes (at each μ value), while the other (*blue*) ablates an equal number of randomly sampled nodes. Both variants show a steep drop-off as μ increases; however, the difference in accuracy between the two is significant. This suggests that ACNs encode important information used by the conv-net for recognition.

3.4. How relevant are ACNs to object recognition?

Farhadi *et al*. [3] show that object recognition using semantic attributes as features is less effective than using hand-crafted features [3]. To evaluate the generality of this result, we study the impact of ACN activations on the task of object recognition in this experiment. To do this, we measure the recognition performance of the pre-trained conv-net before and after ACNs are ablated (i.e. their outputs are manually set to 0). We call this operation 'conv-net surgery', which is similar in spirit but different in purpose to the brain damage strategy proposed by LeCun *et al*. [12], which investigates an efficient way to prune a conv-net without significant performance degradation. In this paper, conv-net surgery is used to probe the relevance of ACN activations in recognizing objects in images. Conceivably, one can perform the reverse operation (i.e. ablate all nodes except for ACNs). However, in this case, the sparsity of ACNs will deactivate the majority of the conv-net and dramatically degrade performance without much meaningful insight.

Quantitative Results: In Figure 8, we report the results of conv-net surgery for various levels of sparsity (controlled by μ) on the validation set of ILSVRC-2012. Clearly, ablating any nodes from the conv-net might decrease accuracy. So, to measure the significance of ACN activations, we require a baseline to compare against. To this end, we perform surgery for two scenarios: **(i)** ablate/inhibit ACNs and **(ii)** ablate/inhibit uniformly random sampled nodes from the conv-net based on the number of selected ACNs on each layer (baseline). Here, ACNs are aggregated from all 25 attributes using an OR operation on their sparse supports. The results of ablating ACNs of individual attributes separately

are shown in the **supplementary material**. Intuitively, the difference in performance between the two schemes should shed light on the inherent relevance of ACNs to object recognition. From the results of Figure 8, it is clear that a slight increase in μ (i.e. increase in the number of ACNs) can lead to a dramatic drop-off in the top-5 accuracy. This drop-off is less severe for the randomized baseline scheme, whose accuracy is averaged over 10 independent runs (standard deviation is also shown), for every value of μ. In some cases, the difference between the two schemes reaches more than 20%. The statistical significance of these differences has been verified numerically using a p-test. For sparsity levels beyond μ=100 (i.e. about 20% of the total number of nodes and more than 80% of the nodes in *fc6* and *fc7*), too many nodes are ablated and discrimination is no longer possible. In summary, these results suggest that ACNs *do* play an important role (not necessarily by themselves) in recognizing objects and that the conv-net *does* mine relations between attributes and objects for recognition.

Qualitative Results: Apart from the effect of ACN ablation on object recognition performance, we investigate the existence of semantic relationships between ablated ACNs of specific attributes and the most (and least) affected object classes. In this case, we perform conv-net surgery using subsets of the 25 attributes and rank the object classes according to their accuracy degradation. Figure 9 shows three such attribute groups and the top-5 most and least affected classes. Interestingly, the most affected classes tend to possess the ablated attributes, while the least affected do not.

4. Conclusion and Future Work

This work shows that there is an intricate relationship between semantic visual attributes and a conv-net trained for object recognition. We empirically show the existence of attribute centric nodes (ACNs) in this conv-net. These nodes encode information that precisely reconstructs attributes in a sparse and unevenly distributed manner among the conv-net layers. We show that ACNs are generalizable and predict attributes better than hand-crafted visual features. We also demonstrate that ACNs are quite important within the conv-net for object recognition. The existence and versatility of ACNs is a stepping stone for developing new algorithms that exploit mid-level concepts in conv-nets.

For future work, we aim to extend our analysis to scene recognition. Using large datasets of attributes and scenes [14, 24], we can extend the applicability of our work, as well as, incorporate other forms of regularization to encourage other priors such as shared activations or correlations between attributes.

Acknowledgments Research reported in this publication was supported by competitive research funding from King Abdullah University of Science and Technology (KAUST). JCN is supported by a Microsoft Research Faculty Fellowship.

Figure 9. Shows object classes that are the most (*red* box) and least (*green* box) affected by ablating ACNs corresponding to three example attribute groups. The mean average precision degradation of each of these classes is reported below its representative image. The most affected classes tend to contain the ablated attributes, while the least affected ones do not. In some cases, the accuracy degradation is tremendous, reaching more than 60%. This is another example of the intricate relationship between ACNs on conv-net and semantic attributes.

References

[1] P. Agrawal, R. Girshick, and J. Malik. Analyzing the performance of multilayer neural networks for object recognition. In *European Conference on Computer Vision (ECCV)*, 2014.

[2] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International Conference on Machine Learning (ICML)*, 2014.

[3] A. Farhadi, I. Endres, D. Hoiem, and D. A. Forsyth. Describing objects by their attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[4] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, Sept 2010.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[6] R. B. Girshick, F. N. Iandola, T. Darrell, and J. Malik. Deformable part models are convolutional neural networks. *CoRR*, abs/1409.5403, 2014.

[7] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding [`https://github.com/BVLC/caffe/tree/master/models/bvlc_alexnet`]. *CoRR*, 2014.

[8] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[10] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[11] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.

[12] Y. LeCun, J. S. Denker, and S. A. Solla. Optimal brain damage. In *Advances in Neural Information Processing Systems (NIPS)*, 1989.

[13] D. Parikh and K. Grauman. Relative attributes. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.

[14] G. Patterson, C. Xu, H. Su, and J. Hays. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108(1-2):59–81, 2014.

[15] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge, 2014.

[16] O. Russakovsky and L. Fei-Fei. Attribute learning in large-scale datasets. In *European Conference of Computer Vision (ECCV), International Workshop on Parts and Attributes*, 2010.

[17] J. Schmidhuber. Deep learning in neural networks: An overview. *CoRR*, abs/1404.7828, 2014.

[18] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: An astounding baseline for recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2014.

[19] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, 2013.

[20] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *European Conference of Computer Vision (ECCV)*, 2012.

[21] E. van den Berg and M. P. Friedlander. Probing the pareto frontier for basis pursuit solutions. *SIAM Journal on Scientific Computing*, 31(2):890–912, 2008.

[22] Y. Wang and G. Mori. A discriminative latent model of object classes and attributes. In *European Conference on Computer Vision (ECCV)*, 2010.

[23] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European Conference of Computer Vision (ECCV)*, 2014.

[24] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.