Object Recognition with and without Objects

Zhuotun Zhu
Computer Science
Johns Hopkins University
zhuotun@gmail.com

Lingxi Xie
Computer Science
Johns Hopkins University
198808xc@gmail.com

Alan L. Yuille
Computer Science
Johns Hopkins University
alan.l.yuille@gmail.com

Abstract

While recent deep neural network models have given promising performance on object recognition, they rely implicitly on the visual contents of the whole image. In this paper, we train deep neural networks on the foreground (object) and background (context) regions of images respectively. Considering human recognition in the same situations, networks trained on pure background without objects achieves highly reasonable recognition performance that beats humans to a large margin if only given context. However, humans still outperform networks with pure object available, which indicates networks and human beings have different mechanisms in understanding an image. Furthermore, we straightforwardly combine multiple trained networks to explore the different visual clues learned by different networks. Experiments show that useful visual hints can be learned separately and then combined to achieve higher performance, which confirms the advantages of the proposed framework.

1. Introduction

Image classification is a long-lasting battle in computer vision, which aims at categorizing an image according to the visual contents. In recent years, we have witnessed an evolution in this research field. Thanks to the availability of large-scale image datasets [5] and powerful computational resources, it is possible to train a very deep convolutional neural network (CNN) [16] for this purpose, which is verified much more efficient beyond the conventional Bag-of-Visual-Words (BoVW) model [3].

It is known that an image contains both foreground and background visual contents. However, most image classification algorithms focus on recognizing the visual patterns only on the foreground region [38]. Although it has been proven that background (context) information also helps recognition [27], it still remains unclear if a deep network can be trained individually to learn visual information from the background region alone. On the other hand, we are interested in exploring the different visual patterns via training neural networks on foreground and background separately for image classification, which is less studied before.

In this work, we investigate the above problems by explicitly training multiple networks for image classification. We first construct datasets from ILSVRC2012 [26], i.e., one foreground set and one background set, by taking advantage of the ground truth bounding box(es) provided in both training and testing cases. After dataset construction, we train deep networks individually to learn foreground (object) and background (context) information, respectively. We find that, even only trained on pure background contexts, the deep network can still converge and makes reasonable prediction (14.4% top-1 and nearly 30% top-5 classification accuracy on the background validation set). We are further interested in the human recognition performance on the constructed datasets. Deep neural networks outperform non-expert humans in fine-grained recognition, and humans sometimes make errors because they cannot memorize all categories of datasets [26]. In this case, to more reasonably compare the recognition ability of humans and deep networks, we follow [13] by merging all the 1,000 fine-grained categories of the original ILSVRC2012, resulting in a 127-class recognition problem meanwhile keeping the number of training/testing images unchanged. We find that human beings tend to pay more attention on the object while networks can put more emphasis on context for classification. By visualizing the patterns captured by the background net, we find that some visual patterns are not available in the foreground net yet help visual recognition. Moreover, we apply networks on the foreground or background regions respectively via the given ground truth bounding box(es) or extracting object proposals without available ones. We find that the linear combination of multiple neural networks can give higher performance.

To summarize, our main contributions are three folds: 1) We demonstrate that learning foreground and background visual contents separately is beneficial for image classification. Training a network based only on background, though looking weird at first glance, is doable and contains highly
useful visual information. 2) We conduct human recognition experiments on either pure background or foreground regions to find that human beings outperform networks on pure foreground while beaten by networks on pure background, which implies the different mechanisms of understanding an image between networks and humans. 3) We straightforwardly combine multiple neural networks to explore the effectiveness of different learned visual clues under two conditions with and without ground truth bounding box(es), which gives comparable improvement over the baseline deep network.

### 2. Related Work

The task of image classification is fundamental in computer vision field, which is aimed to understand the semantic meaning among an image via analyzing its visual contents. Recently, researchers have extended the traditional cases [8][17] to fine-grained [31][21][22], as well as large-scale [35][5][10] tasks. Before the exploding development of deep learning, the dominant Bag-of-Visual-Words (BoVW) model [3] represents every single image with a high-dimensional vector. It is typically composed of three consecutive steps, i.e., descriptor extraction, feature encoding and feature summarization. To avoid the limited descriptive ability of low-level pixels, local descriptors like SIFT [19] and HOG [4] are first extracted. A visual vocabulary is then built to model the data distribution in the feature space. After that, descriptors are quantized onto the vocabulary as compact feature vectors [32][37][23], and aggregated as an image-level description [17]. These feature vectors are post-processed, i.e., normalized, and then fed into a machine learning toolbox [7] for performance comparison with each other.

The Convolutional Neural Network (CNN) is treated as a hierarchical model for large-scale visual recognition. In past years, neural networks have already been proved to be effective for simple recognition tasks [18]. More recently, the availability of large-scale training data (e.g., ImageNet [5]) and powerful computation source like GPUs make it practical to train deep CNNs [16] which significantly outperforms the traditional BoVW models. Specifically, a CNN is composed of numerous stacked layers, in which responses from the previous layer are then convoluted and activated by a differentiable function, sometimes followed by a non-linear transformation [20] to avoid overfitting. Recently, several efficient methods were proposed to help CNNs converge faster [16] as well as prevent overfitting [30][12]. It is believed that deeper networks produce better recognition results [29][27], but also requires engineering tricks to be trained very well [14][11].

Recently, researchers pay more attention to human experiments on objects recognition. Zhou et al. [39] invited Amazon Mechanical Turk (AMT) to identify the concept for segmented images with objects. They found that the CNN trained for scene classification automatically discovers meaningful object patches.

The intermediate responses of CNN, or the so-called deep features map, serve as efficient image description [6], or a set of latent visual attributes. They can be applied to various vision applications, including object detection [9], object parsing [34], image classification [15] and image retrieval [24]. A specific discussion of how different CNN configurations impact the performance is available in [2].

Last but not the least, visualization is an effective method to help understand the mechanism of CNNs. In [38], a de-convolutional operation was proposed to capture visual patterns on different layers of a trained network model. [27] and [11] show that different sets of neurons are activated when a network is used for detecting different visual patterns. The above works are based on a supervised information on the output layer. In this work, we will use a much simpler way of visualization which is inspired by [33].

### 3. Training Networks

Our goal is to explore the possibility and effectiveness of training networks on foreground and background regions, respectively. Here, foreground and background regions are defined by the annotated ground truth bounding box(es) of each image. All the experiments are done on the datasets composed from the ILSVRC2012.

#### 3.1. Data Preparation

The ILSVRC2012 dataset [26] contains about 1.3M training and 50K validation images. Throughout this paper, we refer to the original dataset as OrigSet and the validation images are regarded as our testing set. Among OrigSet, 544,539 training images and all 50,000 testing images are labeled with at least one ground truth bounding box. For each image, there is only one type of object annotated according to its ground truth class label.

We construct three variants of training sets and two variants of testing sets from OrigSet by details below. An illustrative example of data construction is shown in Fig. 1. And the configuration of different image datasets are summarized in Table I.

- The foreground dataset or the FGSet is composed of all images with at least one available ground truth bounding box. For each image, we first compute the smallest rectangle frame which includes all object bounding boxes, then based on which the image inside the frame is cropped to be used as the training/testing data. Note that if an image has multiple object bounding boxes belonging to the same class, we set all the background regions inside the frame to be 0s in order to keep no context information on FGSet. There
Figure 1. Procedures of dataset generation. First we denote the original set as the OrigSet, divided into two sets, one with ground truth bounding boxes and the other one without. Then the set with annotated bounding box(es) are further processed by: setting regions inside all ground truth to be 0s to compose the BGSet while cropping the regions out to produce the FGSet. In the end, add the images without bounding boxes with FGSet to construct the HybridSet. Please note that some images of the FGSet have regions to be black (0s) since these images are annotated with multiple objects belonging to the same class, which are cropped according to the smallest rectangle frame that includes all object bounding boxes in order to keep as less background information as possible on FGSet.

- The construction of the background dataset or the BGSet consists of two stages. First for each image with at least one ground truth bounding box available, regions inside every ground truth bounding box are set to 0s. Chances are that almost all the pixels of one image are set to 0s if its object consists of nearly 100 percent of its whole region. Therefore during training, we discard those samples with less than 50% background pixels preserved, i.e., the foreground frame is larger than half of the entire image, so that we can maximally prevent using those less meaningful background contents (see Fig 1). However in testing, we keep all the processed images, in the end, 289,031 training images and 50,000 testing images are preserved.

- To increase the amount of training data for foreground classification, we also construct a hybrid dataset, abbreviated as the HybridSet. The HybridSet is composed of all images of the original training set. If at least one ground truth bounding box is available, we pre-process this image as described on FGSet, otherwise we simply keep this image without doing anything. As bounding box annotation is available in each testing case, the HybridSet and the FGSet contain the same testing data. Training with the HybridSet can be understood as a semi-supervised learning process.

3.2. Training and Testing

The milestone deep neural network AlexNet [16] is abbreviated as:

\[
C11(S4)@96-MP3(S2)-LRN-C5(P2)@256-MP3(S2)-LRN-C3(P1)@384-C3(P1)@384-C3(P1)@256-MP3(S2)-FC4096-D0.5-FC4096-D0.5-FC1000.
\]

Here, \(C11(S4)@96\) is a convolutional kernel of \(11 \times 11\), a stride length 4 and 96 kernels, and \(P2\) means that the input data is zero-padded with a frame of width 2. \((MP3)S2\) is a max-pooling layer with a window size 3 and a stride length 2. LRN stands for local response normalization [16]. FC4096 is a fully-connected layer with 4096 outputs, and D0.5 is a Dropout layer [28] with randomly 50% responses preserved.

We adopt the same settings as in the CAFFE library [15] and all training images are stored as a LMDB file, and then loaded into the training process in every iteration with 256 samples. The learning rate is set 0.01 at the beginning, and reduced by 1/10 for every 100,000 iterations. The moment is set to be 0.9 and the weight decay parameter is 0.0005. A total of 450,000 iterations are performed, which correspond to around 90 training epochs on the original dataset. Note that both FGSet and BGSet contain less number of images than that of OrigSet and HybridSet, which leads to a larger
number of training epochs, given that the total number of iterations remains unchanged. In these cases, we adjust the dropout ratio as 0.7 to avoid the overfitting issue. We refer to the network trained on the OrigSet as the OrigNet, and similar abbreviated names also apply to other cases, i.e., the FGNet, BGNet and HybridNet.

It is known that data augmentation at the testing stage helps generating higher classification accuracy. Following the conventional approaches [16,27,29], we consider two types of data augmentation, i.e., cropping and flipping. Each testing image is resized to 256 × 256 pixels, while the input size of each trained network is 227 × 227. A typical method is to crop some 227 × 227 sub-images from the testing data, and feed each image (sometimes along with its flipped copy) to the network. Therefore during testing, we report the results by averaging 10 patches from the 5 crops and 5 flips strategy. After all forward passes are done, the average output on the final (fc-8) layer is used for prediction. We adopt the MatConvNet [30] platform for performance evaluation.

4. Experiments

The testing accuracy of AlexNet trained on corresponding dataset are given in the last column of Table 1. We can find that the BGNet produces reasonable classification results: 14.41% top-1 and 29.62% top-5 accuracy (while the random guess gets 0.1% and 0.5%, respectively), which is a bit surprising considering it makes classification decisions only on background contents without any foreground objects given. This demonstrates that deep neural networks are capable of learning pure contexts to infer objects even being fully occluded. Not surprisingly, the HybridNet gives better performance than the FGNet due to more training data available.

### 4.1. Human Recognition

As stated before, to alleviate the possibility of wrongly classifying images for humans beings due to high volume of classes up to 1,000 on the original ILSVRC2012, we follow [13] by merging all the fine-grained categories, resulting in a 127-class recognition problem meanwhile keeping the number of training/testing images unchanged. Details of the 127 classes can be referred to the Supplementary materials. To distinguish the merged 127-class datasets with the previous datasets, we refer to them as the OrigSet-127, FSet-127 and BGSet-127, respectively. Then we invite volunteers who are familiar with the merged 127 classes to perform the recognition task on the BGSet-127 and FSet-127. Humans are given 256 images covering all 127 classes and one image takes around two minutes to make the top-5 decisions. We do not evaluate humans on OrigSet-127 since we believe humans can perform well on this set like on OrigSet.

Table 2 gives the testing recognition performance of human beings and trained AlexNet on different datasets. It is well noted [26] that humans are good at recognizing natural images, e.g., on OrigSet, human labelers achieve much higher performance than AlexNet. We can find the human beings also surpass networks on the foreground (object-level) recognition by 5.93% and 1.96% in terms of top-1 and top-5 accuracy. However, AlexNet beats human labelers to a large margin on the background dataset BGSet-127 considering the 127% and 85% relative improvements from 18.36% to 41.65% and 39.84% to 73.79% for top-1 and top-5 accuracy, respectively. In this case, the networks are capable of exploring background hints for recognition much better than human beings. On the contrary, humans classify images mainly based on the visual contents from the foreground objects. For AlexNet, the performance gaps from OrigSet-127 to BGSet-127 are 43.07% top-1 and 23.89% top-5 relatively smaller drops, which indicates networks potentially put much emphasis on the background information compared with human beings.

| Dataset     | Image Description                  | # Training Image | # Testing Image | Testing Accuracy          |
|-------------|------------------------------------|------------------|----------------|---------------------------|
| OrigSet     | Original Image                     | 1,281,167        | 50,000         | 58.19%, 80.96%           |
| FGSet       | Foreground Image                   | 544,539          | 50,000         | 60.82%, 83.43%           |
| BGSet       | Background Image                   | 289,031          | 50,000         | 14.41%, 29.62%           |
| HybridSet   | Original Image or Foreground Image  | 1,281,167        | 50,000         | 61.29%, 83.85%           |

Table 1. The configuration of different image datasets originated from the ILSVRC2012. The last column denotes for the testing performance of trained AlexNet in terms of top-1 and top-5 classification accuracy on corresponding datasets, e.g., the BGNet gives 14.41% top-1 and 29.62% top-5 accuracy on the testing images of BGSet.

| Dataset     | AlexNet | Human         |
|-------------|---------|---------------|
| OrigSet     | 58.19%, 80.96% | −, 94.90%     |
| BGSet       | 14.41%, 29.62% | −, −          |
| OrigSet-127 | 73.16%, 93.28% | −, −          |
| FSet-127    | 75.32%, 93.87% | 81.25%, 95.83% |
| BGSet-127   | 41.65%, 73.79% | 18.36%, 39.84% |

Table 2. Classification accuracy (in terms of: top-1, top-5) on four image sets by deep neural networks and human, respectively. Human performance on OrigSet (labeled by “∗”) is reported by [26].
Table 3. Cross evaluation accuracy (in terms of top-1, top-5) on four networks and three testing sets. Note that the testing set of HybridSet is identical to that of FGSet.

| Network      | OrigSet          | FGSet            | BGSet            |
|--------------|------------------|------------------|------------------|
| OrigNet      | 58.19%, 80.96%   | 50.73%, 74.11%   | 3.83%, 9.11%     |
| FGNet        | 33.42%, 53.72%   | 60.82%, 83.43%   | 1.44%, 4.53%     |
| BGNet        | 4.26%, 10.73%    | 1.69%, 5.34%     | 14.41%, 29.62%   |
| HybridNet    | 52.89%, 76.61%   | 61.29%, 83.85%   | 3.48%, 9.05%     |

4.2. Cross Evaluation

To study the difference in visual patterns learned by different networks, we perform cross evaluation, i.e., applying each trained network to different testing sets. Results are summarized in Table 3.

We find that the transferring ability of each network is limited, since a model cannot obtain satisfying performance in the scenario of different distributions between training and testing data. For example, using FGNet to predict OrigSet leads to 27.40% absolute drop (45.05% relative) in top-1 accuracy, meanwhile using OrigNet to predict FGSet leads to 7.46% drop (12.82% relative) in top-1 accuracy. We conjecture that FGNet may store very little information on contexts, thus confused by the background context of OrigSet. On the other side, OrigNet has the ability of recognizing contexts but is wasted for the task on FGSet.

4.3. Diagnosis

To fully understand what is inside the networks trained on the background and foreground separately, we conduct several diagnostic experiments to study the property of different networks.

First of all, we report the classification accuracy of different networks with respect to keeping different foreground ratios of the testing image. We split each testing dataset into 10 subsets, each of which contains all images with the foreground ratio no greater than a fixed value. Results are shown in Fig. 2. BGNet gets higher classification accuracy on the images with a relatively small foreground ratio, while all the other three networks prefer a large object since the foreground information is dominant in these cases. The classification accuracy between OrigNet and FGNet or HybridNet is large at the point keeping images of 10% foreground ratio. When the foreground ratio goes larger, e.g., greater than 80%, the performance gap among OrigNet, FGNet and HybridNet gets smaller. It infers that OrigNet trained on OrigSet makes more classification mistakes for images with smaller objects, which are less distinguishable for deep neural networks.

We further investigate the confidence scores distribution of the trained networks by applying them on their corresponding testing sets. According to Table 3, the top-1 classification accuracy of OrigNet, FGNet, BGNet and HybridNet are 58.19%, 60.82%, 14.41% and 61.29%, respectively. For each testing image, we choose the highest prediction score after the softmax operation, and plot the distribution of the highest score over each entire set. Results are shown in Fig. 3. We can see that all networks, except BGNet, are confident in prediction and share the similar distribution, i.e., in more than 50% cases, the highest classification score is higher than 0.5. In comparison, BGNet has less confidence in predicting the background regions of images, and correspondingly, the classification accuracy (14.41%) is much lower than other networks. It indicates that most categories in ImageNet share common background. Therefore, the BGNet cannot as confidently as other networks determine category given only the background regions.

4.4. Visualization

In this part, we visualize the networks to see how different networks learn different visual patterns. We adopt a very straightforward visualization method, which takes a trained network and a set of reference images as input.

We visualize the most significant responses of the neurons on the conv-5 layer. The conv-5 layer is composed of 256 filter response maps, each of which has $13 \times 13$ different spatial positions. After all the 50,000 reference images are processed, we obtain $13^2 \times 50000$ responses for each of the 256 filters. We pick up those neurons with the highest
response and trace back to obtain its receptive field on the input image. In this way we can discover the visual patterns that best describe the concept this filter has learned. For diversity, we only choose at most one patch from a reference image with the highest response score.

Fig. 4 shows visualization results using FGNet on FGSet and BGNet on BGSet, respectively. We can observe quite different visual patterns learned by these two networks. The visual patterns learned by FGNet are often very specific to some object categories, such as the patch of a dog face (filter 1) or the front side of a shop (filter 3). These visual patterns correspond to some visual attributes, which are critical for recognition. On the other side, each visual concept learned by BGNet tends to appear in many different object categories. These visual patterns are often found in the context, which play an assistant role in object recognition. Although some of them (e.g., filter 20: a portrait in a suit) may also be quite specific, we point out that these categories (e.g., tie) often have a quite small foreground ratio, thus it is easy to associate the object category to other high-correlated objects (person) even when the foreground contents are not available.

To summarize, FGNet and BGNet learn different visual patterns that can be combined to assist visual recognition. In Section 4.3 we quantitatively demonstrate the effectiveness of these networks via combining these information for better recognition performance.

5. Combination

We first show that the recognition accuracy can be significantly boosted using ground truth bounding box(es) at the testing stage. Next, with the help of the EdgeBox algorithm [40] to generate accurate object proposals, we improve the recognition performance without the requirement of ground truth annotations. We name them as guided and unguided combination, respectively.

5.1. Guided vs. Unguided Combination

We first describe guided and unguided manners of model combination. For simplicity, we adopt the linear combination over different models, i.e., forwarding several networks, and weighted summing up the responses on the fc-8 layer.

If the ground truth bounding box is provided (the guided condition), we use the ground truth bounding box to divide the testing image into foreground and background regions. Then, we feed the predicted foreground regions into FGNet or HybridNet, and predicted background regions into BGNet, then fuse the neuron responses at the final stage.

We also explore the solution of combining multiple networks in an unguided manner. As we will see in Section 5.2, a reliable bounding box helps a lot in object recognition. Motivated by which, we use an efficient and effective algorithms, the EdgeBox, to generate a lot of potential bounding boxes proposals for each testing image, and then feed the

![Figure 2](image2.png)

Figure 2. Classification accuracy with respect to the foreground ratio of testing images. The number at, say, 0.3, represents the testing accuracy on the set of all images with foreground ratio no greater than 30%.

![Figure 5](image5.png)

Figure 5. Edgebox statistics on ImageNet validation set, which denotes the curriculum distribution function of the detected ground truth with respect to the top-k proposals. Here, we set the Intersection of Union (IOU) threshold to be 0.7 for EdgeBox algorithm.
Figure 4. Patch visualization results of **FGNet** on **FGSet** (above) and **BGNet** on **BGSet** (below). This figure is best viewed in color PDF. Each line is corresponded to one filter on the conv-5 layer, and each patch is selected from $13^2 \times 50000$ ones, with the highest response on that kernel. We label the abbreviated category name above each patch, and the full name can be found in the Supplementary materials. Note that **FGNet** learns quite specific visual patterns which are highly relevant to concrete visual attributes, while the filters learned by **BGNet** can fire at many different categories and play an assistant role in recognition.
foreground and background regions into neural networks as described before across top proposals.

First of all, we demonstrate the EdgeBox proposals are good to capture the ground truth object. After extracting top-k proposals with EdgeBox, we count the detected ground truth if at least one of proposals has the IoU no less than 0.7 with the ground truth. The cumulative distribution function (CDF) is plotted in Fig. 5. From this figure, choosing the top-300 bounding boxes is sufficient to cover nearly 90% objects; when the number of proposals is decreased to 100, due to the consideration of computational overhead, the recall goes down to about 81%, a little bit lower but still acceptable. Considering efficiency as well as accuracy, we feed foreground and background defined by the top-100 proposals into the trained networks. After obtaining 100 outputs for each network, we average the responses on the fc-8 layer for classification.

5.2. Results and Discussion

Results of different combinations are summarized in Table 4. Under either guided or unguided settings, combining multiple networks boosts recognition performance, which verifies the phenomenon that different visual patterns from different networks can help with each other for the object recognition.

For the guided way of testing, by providing accurate separation of foreground from background, works better than the unguided way by a large margin. Then first take a closer look at the accuracy gain under the unguided condition. The combination of HybridNet + BGNet outperforms HybridNet by 2.5% and 2.47% in terms of top-1 and top-5 recognition accuracy, which are noticeable gains. As for the FGNet + BGNet, it improves 1.12% and 1.20% classification accuracy compared with the FGNet, which are promising. Surprisingly, the combination of HybridNet with OrigNet can still increase from the OrigNet by 2.68% and 1.61%. We hypothesize that the combination is capable of discovering the objects implicitly by inference of where the objects are due to the visual patterns of HybridNet are learned from images with object spatial information. These improvements are super promising considering that the networks don’t know where the accuracy objects are during under the unguided condition. Notice that the results under unguided condition cannot surpass those under guided condition, arguably because the top-100 proposals not good enough to capture the accurate ground truth given that the BGNet cannot give high confidence on the predictions.

As for the guided condition, improvements can consistently be found after combinations with the BGNet. Specifically, HybridNet + BGNet gets 1.23% and 0.68% accuracy enhancement in terms of top-1 and top-5. It is well worth noting that the combination of HybridNet with OrigGNet improves the baseline of OrigGNet to a significant margin by 7.44% and 5.37%. The huge gains in this situation makes sense accounting for networks’ ability of inferring object locations trained on the accurate bounding box(es).

6. Conclusions and Future Work

In this work, we first demonstrate the surprising finding that neural network can predict object categories quite well even when the object is not present. This motivates us to study the human recognition performance for foreground objects and background context without objects. We show on the 127-classes ILSVRC2012 that human beings beat neural networks for foreground object recognition, but perform much worse to predict the object category only on the background without objects. Then we show combining the visual patterns learned from different networks can help each other for the recognition task. We conclude that more emphasis should be placed on the role of context for object detection and recognition.

In the future, we will investigate an end-to-end training algorithm for explicitly separating and then combining the foreground and background information, which explores the visual contents to the full extent. For instance, inspired by some joint learning strategy such as the Faster R-CNN [25], we can design a structure which predicts the

| Network          | Guided Combination | Unguided Combination |
|------------------|--------------------|----------------------|
| OrigNet          | 58.19%, 80.96%     | 58.19%, 80.96%       |
| BGNet            | 14.41%, 29.62%     | 8.30%, 20.60%        |
| FGNet            | 60.82%, 83.43%     | 40.71%, 64.12%       |
| HybridNet        | 61.29%, 83.85%     | 45.58%, 70.22%       |
| FGNet+BGNet      | 61.75%, 83.88%     | 41.83%, 65.32%       |
| HybridNet+BGNet  | 62.52%, 84.53%     | 48.08%, 72.69%       |
| HybridNet+OrigNet| 65.63%, 86.69%     | 60.84%, 82.56%       |

Table 4. Classification accuracy (in terms of: top-1, top-5) comparison of different network combinations. It’s worth noting that we feed the entire image into the OrigNet no matter whether the ground truth bounding box(es) is given in order to keep the testing phase consistent with the training of OrigNet. Therefore, the reported results of OrigNet are same with each other under both guided and unguided conditions. To integrate the results from several networks, we weighted sum up the responses on the fc-8 layer.
object proposals in the intermediate stage, then learns the foreground and background regions derived from the proposals separately by two sub-networks and then takes foreground and background features into further consideration.

References

[1] C. Cao, X. Liu, Y. Yang, Y. Yu, J. Wang, Z. Wang, Y. Huang, L. Wang, C. Huang, and W. Xu. Look and Think Twice: Capturing Top-Down Visual Attention with Feedback Convolutional Neural Networks. ICCV, 2015.

[2] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the Devil in the Details: Delving Deep into Convolutional Nets. BMVC, 2014.

[3] G. Csurka, C. Dance, L. Fan, Willamowski, and C. Bray. Visual Categorization with Bags of Keypoints. Workshop on ECCV, 2004.

[4] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. CVPR, 2005.

[5] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. CVPR, 2009.

[6] 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. ICML, 2014.

[7] R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin. LIBLINEAR: A Library for Large Linear Classification. Journal of Machine Learning Research, 9:1871–1874, 2008.

[8] L. Fei-Fei, R. Fergus, and P. Perona. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. Computer Vision and Image Understanding, 96(6):411–418, 2004.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. CVPR, 2014.

[10] G. Griffin, A. Holub, and P. Perona. Caltech-256 Object Category Dataset. Technical Report: CNS-TR-2007-001, Caltech, 2007.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. CVPR, 2016.

[12] G. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Improving Neural Networks by Preventing Co-adaptation of Feature Detectors. arXiv preprint, arXiv: 1207.0580, 2012.

[13] M. Huh, P. Agrawal, and A. Efros. What Makes ImageNet Good for Transfer Learning? arXiv: 1608.08614, 2016.

[14] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. ICML, 2015.

[15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. CAFFE: Convolutional Architecture for Fast Feature Embedding. ACM International Conference on Multimedia, 2014.

[16] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. NIPS, 2012.
[36] L. Xie, J. Wang, Z. Wei, M. Wang, and Q. Tian. DisturbLabel: Regularizing CNN on the Loss Layer. *CVPR*, 2016.

[37] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear Spatial Pyramid Matching Using Sparse Coding for Image Classification. *CVPR*, 2009.

[38] M. Zeiler and R. Fergus. Visualizing and Understanding Convolutional Networks. *ECCV*, 2014.

[39] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object Detectors Emerge in Deep Scene CNNs. *ICLR*, 2015.

[40] C. Zitnick and P. Dollar. Edge Boxes: Locating Object Proposals from Edges. *ECCV*, 2014.
### Supplemental Materials

#### A. List of 127 Classes

In Section 4.1 we follow [13] to merge the original 1K classes of ILSVRC2012 to 127 classes for human recognition experiment. The list below shows all the 127 classes mentioned on the OrigSet-127, FSet-127 and BGSet-127.

- n02856463 board
- n02638596 ganoid
- n04235291 sled
- n01248580 soft-finned fish
- n10401829 participant
- n04500060 turner
- n02519238 food fish
- n03664943 ligament
- n03446832 golf equipment
- n03764276 military vehicle
- n03122748 covering
- n06874019 light
- n03619396 kit
- n04128837 sailing vessel
- n03528263 home appliance
- n04100174 rod
- n03880531 pan
- n02924116 bus
- n10493783 chameleon
- n04152044 protective garment
- n01482330 shark
- n07692318 sign
- n01629276 salamander
- n13134947 fruit
- n02954340 cap
- n09820863 athlete
- n09724060 bar
- n01687665 agamid
- n03476083 hairpiece
- n03241093 drill rig
- n01351500 cushion
- n10703569 ceratopsian
- n01861778 mammal
- n04350516 gown
- n01495701 ray
- n0367362 litter
- n04371292 weight
- n03597469 jewelry
- n04077734 rescue equipment
- n03294833 eraser
- n01900942 coelenterate
- n07929519 coffee
- n04285622 sports implement
- n03497657 hat
- n07930554 punch
- n01692864 lacerial lizard
- n07683786 loaf of bread
- n0309947 cleaning implement
- n04447443 toiletty
- n02642644 scorpænid
- n10019552 diver
- n01726692 snake
- n02942699 camera
- n03174202 stick
- n01767661 arthropod
- n01674990 gecko
- n0339808 fabric
- n03257866 duplicator
- n07829412 sauce
- n01922303 worm
- n01940736 mollusk
- n07882497 coccincoication
- n02605316 butterfly fish

#### B. List of Abbreviated Category Names

In Fig. 4 we illustrate the patches that give the top highest responses for each filter on layer conv-5 from the all the reference images. The following list gives the full category names in alphabet ascending orders of the abbreviated ones in the figure.

- AfrGray: American grey
- Barbell: barbell
- Blowfish: blowfish
- BowTie: bow tie
- CandSto: candy store
- CarWhe: car wheel
- Chute: parachute
- Coast: coast
- Colob: colobus
- Daisy: daisy
- EntCent: entertainment center
- Gond: gondola
- Hamst: hamster
- Hornbill: hornbill
- IndBlu: indigo bunting
- Jetty: jetty
- LabCoa: lab coat
- Leopard: leopard
- Lionfish: lionfish
- Lyca: lycaenid
- Mali: malinois
- MealO: meat loaf
- Nautil: nautilus
- Panda: panda
- Peacock: peacock
- Pinwhe: pinwheel
- Pomeg: pomegranate
- Pret: pretzel
- Rab: Angora rabbit
- Rai: rain barrel
- Redsh: redshank
- Ringlet: ringlet
- SeaUrch: sea urchin
- Slug: slug
- Spaniel: Japanese spaniel
- Straw: strawberry
- SweShir: sweater shirt
- Thru: throne
- Trad: traffic light
- Troll: trolleybus
- Viad: viaduct
- Windsor: tie
- Antich: artichoke
- BarShop: barbershop
- BookShop: bookshop
- Bulb: bulb
- Cardoon: cardoon
- Chetah: cheetah
- Cinema: cinema
- Coil: coil
- Cuke: cucumber
- Dome: dome
- Goldfish: goldfish
- GroSto: grocery store
- Hook: hook
- HowMon: bowler monkey
- Jaguar: jaguar
- Kuvasz: kuvasz
- Lemon: lemon
- Lhasa: lhasa
- Lolly: lolly
- Mallot: mallot
- Maltese: Maltese dog
- MixFlow: mixing bowl
- Pakni: paper knife
- Papillon: papillon
- PillBottle: pill bottle
- Pizza: pizza
- PowDri: power drill
- Puma: puma
- RadTel: radio telescope
- Rapes: rapseed
- RelfCam: reflex camera
- Ruler: ruler
- Shih-Tzu: Shih-Tzu
- SpaShu: space shuttle
- SwCa: swimming cap
- Terrier: Tibetan terrier
- Tobah: tobacco shop
- Trim: trimaran
- Vault: vault
- Website: website
- Wom: wombat

#### C. Further Thoughts on Human Experiments

That machine learning algorithms can outperform humans on some visual tasks is sometimes an artifact of the datasets that these claims are tested on. A machine learning algorithm can ace the test because it exploits the assumption that the test questions must be similar to those it has been trained on. By comparison, human visual systems are designed to perform an enormous range of tasks on this universal distribution of images and so cannot, without significant training, take advantage of the regularities in the specific dataset being tested on. Moreover, human visual abilities are known to be adaptive to specific types of tasks and visual environments, e.g., an expert radiologist can detect tumors that appear invisible to a normal human. In summary, human abilities are general purpose, adaptive, and flexible while machine learning methods remain highly specialized.