Visualization of Deep Convolutional Neural Networks

Dingwen Li
Washington University in St Louis

Follow this and additional works at: http://openscholarship.wustl.edu/eng_etds

Part of the Engineering Commons, and the Other Computer Sciences Commons

Recommended Citation
Li, Dingwen, "Visualization of Deep Convolutional Neural Networks" (2016). Engineering and Applied Science Theses & Dissertations. 150.
http://openscholarship.wustl.edu/eng_etds/150

This Thesis is brought to you for free and open access by the Engineering and Applied Science at Washington University Open Scholarship. It has been accepted for inclusion in Engineering and Applied Science Theses & Dissertations by an authorized administrator of Washington University Open Scholarship. For more information, please contact digital@wumail.wustl.edu.
Visualization of Deep Convolutional Neural Networks

by

Dingwen Li

A thesis presented to the Graduate School of Arts and Sciences of Washington University in partial fulfillment of the requirements for the degree of

Master of Science

May 2016
Saint Louis, Missouri
copyright by
Dingwen Li
2016
Contents

List of Tables ......................................................... iii
List of Figures ....................................................... iv
Acknowledgments ................................................... v
Abstract ................................................................. vii

1 Introduction ......................................................... 1

2 Visualization Approaches ........................................ 4
  2.1 Analysis by Image Masking ..................................... 5
    2.1.1 Existing Work ............................................. 5
    2.1.2 Application to Our Tasks ................................. 6
  2.2 Analysis by Taylor Decomposition ............................ 7
    2.2.1 Existing Work ............................................. 7
    2.2.2 Our Implementation ..................................... 9
  2.3 Analysis by Inception ......................................... 11
    2.3.1 Existing Work ............................................. 12
    2.3.2 Our Modification ....................................... 13

3 Experimental Results and Applications ........................ 18
  3.1 Visualizing Time of Year ...................................... 18
    3.1.1 Visualization by Image Masking ......................... 19
    3.1.2 Visualization by Taylor Decomposition ................. 21
  3.2 Visualizing Wheat Heading .................................... 22
    3.2.1 Visualization by Image Masking ......................... 24
    3.2.2 Visualization by Taylor Decomposition ................. 25
    3.2.3 Visualization by Inception ............................ 25

4 Conclusion ......................................................... 51

References ............................................................ 53

Vita .......................................................... 56
List of Tables

3.1 Visualization Approaches Used for Each Model . . . . . . . . . . . . . . . . . 23
List of Figures

2.1 Convolutional Neural Network Architecture .......................... 5
2.2 Relevance Map ..................................................... 10
2.3 Visualization of Cat by Inception .................................. 14
2.4 Visualization of April ................................................ 16
2.5 The Visualization for Wheat Heading ............................... 17

3.1 Image Calender of Golf Course ..................................... 27
3.2 Visualization by Image Masking, Golf Course ....................... 28
3.3 Image Calender of Mediterranean Island ............................ 29
3.4 Visualization by Image Masking, Mediterranean Island .......... 30
3.5 Image Calender of Superior National Forest ....................... 31
3.6 Visualization by Image Masking, Superior National Forest ...... 32
3.7 Image Calender of Sequoia National Park ......................... 33
3.8 Visualization by Image Masking, Sequoia National Park ......... 34
3.9 Image Calender of Half Dome ....................................... 35
3.10 Visualization by Image Masking, Half Dome ...................... 36
3.11 Visualization by Taylor Decomposition, Golf Course .......... 37
3.12 Visualization by Taylor Decomposition, Mediterranean Island 38
3.13 Visualization by Taylor Decomposition, Superior National Forest 39
3.14 Visualization by Taylor Decomposition, Sequoia National Park 40
3.15 Visualization by Taylor Decomposition, Half Dome, Yosemite 41
3.16 Visualization by Image Masking, 0 Wheat Heading, Regression Model 42
3.17 Visualization by Image Masking, 50 Wheat Heading, Regression Model 43
3.18 Visualization by Image Masking, 100 Wheat Heading, Regression Model 44
3.19 Visualization by Image Masking, 0 Wheat Heading, Classification Model with 2 Outputs, Training from Scratch ................................. 45
3.20 Visualization by Image Masking, 100 Wheat Heading, Classification Model with 2 Outputs, Training from Scratch ................................. 46
3.21 Visualization by Taylor Decomposition, Regression Model ...... 47
3.22 Visualization by Taylor Decomposition, Classification Model .... 48
3.23 Visualization by Inception, Classification Model with 2 Outputs, Training from Scratch ........................................... 49
3.24 Visualization by Inception, Classification Model with 2 Outputs, Fine Tuning 49
3.25 Visualization by Inception, Classification Model with 219 Outputs, Fine Tuning 50
3.26 Visualization by Inception, Classification Model with 1002 Outputs, Fine Tuning 50
Acknowledgments

I would like to express my gratitude to my thesis advisor Professor Robert Pless for the continuous support of my thesis study and research. He always gives me guidance and suggestions when I get stuck in some research problems. Also, he inspire me how to think logically and rigorously.

Dingwen Li

Washington University in Saint Louis
May 2016
Dedicated to my parents.
Deep learning has achieved great accuracy in large scale image classification and scene recognition tasks, especially after the Convolutional Neural Network (CNN) model was introduced. Although a CNN often demonstrates very good classification results, it is usually unclear how or why a classification result is achieved. The objective of this thesis is to explore several existing visualization approaches which offer intuitive visual results. The thesis focuses on three visualization approaches: (1) image masking which highlights the region of image with high influence on the classification, (2) Taylor decomposition back-propagation which generates a per pixel heat map that describes each pixel’s effect on the classification, and (3) Inception which generates a natural looking image based on the features maximizing the classification score. We explore two challenging visualization tasks, (1) visualizing a model classifying images based on the time when they are taken, and (2) visualizing a model of predicting plant phenotypes (specifically wheat heading percentage). The thesis demonstrates how these visualization approaches work for both the classification model and regression model, and evaluates the results on real-world imagery.
Chapter 1

Introduction

The Convolutional Neural Network (CNN) is a state of the art deep learning platform for large scale image classification tasks. CNN has been used as a primary tool for image classification, since a GPU-based implementation of CNN was introduced [2] [3]. CNNs overwhelm both other multi-layer perceptrons and alternative approaches on the accuracy for large scale data sets. Additionally, they have relatively simple implementation and fast computation at test time. There are a few very common CNN architectures that have been proposed, such as AlexNet [8], GoogleNet [19] and VGG-Net [17]. These architectures press the limits of what can fit into a GPU memory and acceptable training time to maximize classification accuracy. Nowadays, carefully designed CNN can even outperform humans in image classification and pattern recognition.

Although CNN can achieve such great performance, it acts like a black box, and we sometimes do not understand how these networks make decisions and what information is stored by them. One way to shed light on the CNN is to visualize the network, by generating images or visual results that can in some extent reflect what information is stored by CNN. These results could be guidelines for designing networks with higher accuracy, and could give insight into whether they may be effective when tested on new datasets.

Several visualization methods have been proposed in recent years. Firstly, the filter visualization approach has been proposed by Erhan et al. in their work, as a way to see what information has been stored in the weights of deep network. Then Zeiler and Fergus came up with a novel visualization architecture call Deconvnet [21]. Deconvnet maps activations
in the intermediate layers back to input pixel space, showing which input patterns or features cause the corresponding activations. Also in their work, a sliding window occlusion approach has been introduced. This creates a heat map visualizing the parts of the images that most affect the classification. Probably, the most popular CNN visualization method is called Inceptionism [12], which is a gradient-based reconstruction approach introduced by Google. The gradient-based reconstruction approach can help us understand what is encoded in a CNN by inverting their deep representation [9]. This work by activation maximization, starting with a base image and generating images that make the most strongly activated features even strong [10]. Recently, Inceptionism demonstrates another powerful aspect in visualization by uncovering multiple types of features learned by each neuron in deep neural network [14].

Besides the above visualization approaches, there are also some trying to interpret CNN in other ways. Zhou et al. propose a visualization that masks out irrelevant region in images to accentuate the significant region based on the actual receptive field and feature map [23]. Recently, Relevance Map Propagation and Taylor Decomposition have been introduced by Bach et al. as novel approaches to visualize CNN by heat maps [1]. Later, an approach combining Relevance Map Propagation and Taylor Decomposition came out as Explaining NonLinear Classification Decisions with Deep Taylor Decomposition [11], which is a novel way to decompose the final activation function and showing pixel-wise CNN visualization. The final outcome will be viewed as heat map, which gives a better explanation of CNN’s decision making than other heat map visualization approaches [16], such as sensitivity-based approach [18] or the deconvolution method [21].

This thesis will discuss three visualization approaches, which have their own perspectives on interpreting CNN classification decision. I focus on my implementation and modification for these approaches. The following sections will be organized as follows. First, the analysis of each visualization approach, then my implementations or modifications based on the existing works, and finally what we have learned from these approaches.

Experiments are performed on two types of classification task, visualizing time of year with webcam data from AMOS [6] and visualizing wheat heading – which classifies pictures of wheat crops based on how much of their seeds have appeared. These experiments will demonstrate how I use these approaches to visualize the CNN. The reason why I am choosing these
two classification tasks is that their categories are difficult to distinguish in some sense. Since webcams are fixed, there is no obvious difference among categories in the time of year model, except some season changes. For the wheat heading model, each category represents images with same heading percentage but the wheat headings are spread in various locations and orientations. So it is meaningful for us to know what factors play an important role to make the classification decisions. Finally, a conclusion will be drawn on our contributions to CNN visualization, what is the difference between these approaches, and in what cases they should be applied in order to have a reasonable result.
Chapter 2

Visualization Approaches

In this chapter we will present three visualization approaches. Image Masking based on feature map and Taylor Decomposition Back-Propagation are the visualization methods that both generate results by analyzing activation function. Image masking focuses on finding the region that contributes the most to the activation function, and Taylor decomposition back-propagation finds out the relevance between pixels and the activation function. Visualization by Inception, introduced by Google in its research blog as [12], is now a popular way of generating painting-like images from random noise, which contain important information stored by CNN.

The CNN models we used here share the similar 8-layer architecture as AlexNet [8]. They are either classification model or regression model, trained from scratch or fine tuned from AlexNet, the basic architecture of our models used in the study is showed as Figure 2.1. Note here, the pooling layers and the rectified linear layers are omitted in this figure, actually, they are exist in our models. These models basically have the same architecture, except the last layer with different output numbers. One difference in architecture between our model and AlexNet model is that we perform training and testing on a single GPU, rather than split the tasks and put them onto two GPUs, which is the design used in the original AlexNet implementation [8]. This slightly difference will make it easier to analyze the inner layers. The reason I am using AlexNet as a pre-trained model is that it is a widely used CNN model with high accuracy. In the following, I will give an analysis for existing work of each approach and then our modification for applying these methods to our cases.
Figure 2.1: Convolutional neural network architecture used in our experiments, pooling layers and relu layers are omitted in this figure

2.1 Analysis by Image Masking

Convolutional neural network visualization by image masking is a way to observe directly which parts of a image have significant influence on the final activation score. This approach is mainly applied for the situation of interpreting the decision making in scene classification tasks, where many objects with semantic meaning are contained in an image. In this section, I will demonstrate how this approach can be applied for visualizing CNN model of scene classification from our AMOS webcam data set and whether it can be used to visualize CNN for wheat heading task.

2.1.1 Existing Work

The image masking approach to CNN visualization has been introduced by Zhou, et al. [23]. They introduce this method to observe which objects emerge as going from the bottom layer to the upper layer, when doing scene classification. In their work, the image masking method is performed by segmenting input images and masking out irrelevant region with activations below certain threshold based on feature map of a specific unit in a specific layer. The
neurons with maximum activation in a feature map are first determined and then a mask is applied to this image to mask out region that has activation less than a certain threshold relative to the maximum activation. Usually, the threshold value is the maximum activation times a constant, \( c < 1 \), which may be different for each model, in order to accentuate the region with high activation.

This approach is an image-centric visualization, since each visualization is generated based on a specific image and the visualization results are showed on that image. Also, image masking visualization allows us to do some statistics using intermediate results. Interesting statistics might include the frequency of objects emerging as a high activation region, the counts of CNN units discovering each object class and the frequency of most informative objects for scene classification [23].

2.1.2 Application to Our Tasks

The image masking based on feature map gives an intuitive demonstration of which part of an image or which objects in a scene has significant contribution for final activation score. Inspired by their work, I implemented a modification of their approach to apply to our visualization of CNN model for predicting time of year on AMOS data and wheat heading.

In my implementation, all the results are from the conv5 layer, an upper level convolutional layer, since high level features, like objects, tend to emerge in these upper levels [20]. We get a more intuitive observation of high activation region by highlighting objects, which are high level semantics, compared to low level features generated by the bottom layers. Conv5 totally contains 256 units with 256 feature maps (the structure of conv5 layer showed as Figure 2.1), but some of the feature maps have zero total activation score. The feature map with zero anywhere will not generate any visualization, since there is no maximum value to determine the threshold value. So I choose the top 3 units with high activation score for each image forwarding through the network and generate visualization results for each of these 3 units. Then I select images with the highest final activation score from each category to form a visualization for the whole CNN model. In my experiment, I apply this approach to visualize the most "active" object for each month category, which gives us an
intuitive idea about how CNN makes decision and how important an object can affect the scene classification result. The experimental results will be showed on Section 3.1.1. Also the experiments for applying image masking to wheat heading CNN visualization will be showed on Section 3.2.1.

2.2 Analysis by Taylor Decomposition

For both classification model and regression model, the final decision making is based on the activation function. Since the final activation function is a result of summing all the activation score from previous neurons, there is an idea to directly decompose this final activation function as Taylor Series and then back-propagate these series from the last layer down to the first input layer. A heat map can be generated to show per pixel relevance, which indicates the pixel’s relevance with the final activation score. If the relevance is high for some pixels, it means that these pixels account for high contribution for the final classification decision.

2.2.1 Existing Work

Pixel-wise decomposition approach for visualizing non-linear models has been proposed by Bach et al. [1] as either relevance propagation or Taylor decomposition. Later Montavon et al. reconcile these two decomposition approaches as a Deep Taylor Decomposition framework specifically for Deep Neural Networks [11].

The Deep Taylor Decomposition framework is built on two basic facts. First, the relevance score associated to each neuron (or pixel in the first layer) is conservative in the process of back-propagation, which means the sum of assigned relevance for each neuron/pixel stay the same for all the layers:

\[
 f(x) = \ldots = \sum_{d \in (l+1)} R_{d}^{l+1} = \sum_{d \in (l)} R_{d}^{l} = \ldots = \sum_{p} R_{p}
 \] (2.1)
where \( f(x) \) is the final activation score, \( R_d^l \) is the relevance for \( d \) neuron in layer \( l \), \( R_p \) is the relevance associated to input image pixel.

Second, the first-order Taylor expansion of a real value function is given as:

\[
f(x) = f(\tilde{x}) + \left( \frac{\partial f}{\partial x} \bigg|_{x=\tilde{x}} \right)^T \cdot (x - \tilde{x}) + \varepsilon
\]

(2.2)

where \( \tilde{x} \) is a root of \( f(x) \), \( \varepsilon \) denotes Taylor residual. The above expansion can be reduced if a root \( \tilde{x} \) is chosen, which makes \( f(\tilde{x}) = 0 \), and we approximate the activation function by first-order Taylor expansion, Equation 2.3,

\[
f(x) = \left( \frac{\partial f}{\partial x} \bigg|_{x=\tilde{x}} \right)^T \cdot (x - \tilde{x})
\]

(2.3)

for purpose of easy implementation and analysis without losing much accuracy.

Assuming the root is chosen, and the final activation is approximated by the first-order Taylor expansion, we apply the conservation rule, Equation 2.1, to the first-order Taylor expansion, Equation 2.3, then the propagation equation can be written as

\[
\sum_i R_i = \sum_i \left( \frac{\partial \sum_j R_j}{\partial x_i} \bigg|_{\{\tilde{x}_i\}} \cdot (x_i - \tilde{x}_i^{(j)}) \right)
\]

(2.4)

\[
R_i = \sum_j \frac{\partial R_i}{\partial x_i} \bigg|_{\{\tilde{x}_i\}} \cdot (x_i - \tilde{x}_i^{(j)})
\]

(2.5)

where \( R_i \) is the relevance in bottom layer, \( R_j \) is the relevance in top layer, \( \tilde{x}_i^{(j)} \) indicates the roots for corresponding relevance \( R_j \). The above equation is the general form of Taylor decomposition back-propagation, so the exact formula will differ according to the roots \( \tilde{x}_i^{(j)} \) chosen when in different input domain.

Montavon et al. propose \( w_2 \)-Rule and \( z \)-Rule to explicitly associate the relevance propagation rule with the weights in CNN. \( w^2 \)-Rule is used for unconstrained input space, and \( z \)-Rule is used for constrained input space [11]. For example, the network contains rectified linear units restricting the input in the back-propagation which is the output in the forward pass to be greater than 0, or back-propagation arrives at the first layer where the input is the
pixel value, ranging from 0 to 255. Equation 2.6 is the propagation rule for unconstrained input space, namely the $w^2$-Rule:

$$R_i = \sum_j \frac{w_{ij}^2}{\sum_{i'} w_{i'j}^2} R_{j}$$  \hspace{1cm} (2.6)$$

where $w$ is the weights of filter. Equation 2.7 and 2.8 are the propagation rule for constrained input space. Equation 2.7 is called $z^+$-Rule, which is used for the case of input $x > 0$.

$$R_i = \sum_j \frac{z_{ij}^+}{\sum_{i'} z_{i'j}^+} R_{j}$$  \hspace{1cm} (2.7)$$

where $z_{ij}^+ = x_i w_{ij}^+$, $w_{ij}^+$ denotes the positive part of weights. Equation 2.8 is called $z^\beta$-Rule, which is applied for the case that constrained by domain $\{\{x_i\} : \forall_{i=1}^d l_i \leq x_i \leq h_j\}$.

$$R_i = \sum_j \frac{z_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-}{\sum_{i'} z_{i'j} - l_i w_{i'j}^+ - h_i w_{i'j}^-} R_{j}$$  \hspace{1cm} (2.8)$$

where negative sign denotes negative part of weights.

After obtaining the pixel-wise relevance map for the input image, a heat map with the color intensity indicating the relevance of each pixel with the final activation will be generated, Figure 2.2.

### 2.2.2 Our Implementation

The CNN models I used here are classification model and regression model which have the similar architecture as AlexNet with image classification task. In order to make the Taylor expansion exist, namely there is a root for $f(x_0) = 0$ in the input domain, a proper rule should be chosen. The input domains for the intermediate layers are just $[0, +\infty)$ because of the exist of rectified linear units, which corresponds to $z^+$ rule, and for the first layer, input data, the range is $[0, 255]$, which corresponds to $z^\beta$ rule.
For a CNN, the number of neurons in the intermediate layers is large. Thus, the computational complexity when evaluating the relevance propagation is enormous, $O(n^2)$, $n$ is the number of neurons in each layer, for example, 290400 neurons in the second layer and 154587 neurons in the first layer. Instead of directly computing two loops of summation, the computation complexity can be reduced by applying matrix multiplication and the characteristics of CNN’s sparse connectivity, locality and weights sharing. The matrix arithmetic form of Equation 2.7 is

$$R_i = (W_p^T \cdot (R_j \oslash (W_p \cdot X^T))) \odot X$$ (2.9)

where $W_p$ denotes positive part of weights, $\oslash$ is component-wise matrix division and $\odot$ is component-wise matrix multiplication. Also, the original 4-dimensional inputs and weights matrix can be flatten to 2-dimensional matrix, which can save a lot amount of computation time. Since weights are shared between neurons among the same depth, the dimension of the matrix can be further reduced by the common index used in different filters. When back-propagating through the pooling layers, a method called unpooling [15] [21] [22] is used to restore the feature map information before the pooling layer.

Figure 2.2: An example of relevance map generate by Taylor decomposition
In my implementation, the positions of maximum neurons selected in the max-pooling layer are stored during the forward pass. So when performing back-propagation, the neurons with max value will be retrieved and other neurons will be restored as 0. Since we are finding the pixels with high relevance in the data layer, use 0 to represent relevance for other neurons that have been pooled out in the forward pass will not affect the final outcome.

I apply this method to both classification model and regression model of visualizing time of year and wheat heading. In addition to showing the heat map for per pixel relation with final activation, I make a more vivid visualization. Firstly, a gray image of the original image is generated. Then, we add red color with different intensity to the pixels in gray level images where the relevance is above a certain threshold. The intensity of red indicates the relevance of this pixel with the final activation score. This approach will accentuate the region of high relevance. Several images from different categories are selected as demonstrating the result of visualization. Details about experiment and the visualization results will be showed in the next Chapter.

### 2.3 Analysis by Inception

The above two visualization approaches are based on analyzing activation score and generating results directly on the input image. So these visualization approaches aim at showing which parts of image have important influence on the final classification result. In Inception approach, which is also called activation maximization in the work of Mahendran and Vedaldi [10], a reconstruction of the input image from image representation will be showed as the visualization of CNN. The information used to reconstruct the input image is gathered entirely from the CNN. So this method is a visualization from the perspective of visualizing what information have been stored in CNN in order to achieve some extent of accuracy.
2.3.1 Existing Work

In recent years, some Inception approaches and their variants have achieved great results for being natural looking and resembling specific objects in the input image. All these methods are built on the gradient-based approaches with regularization terms added. Visualization by activation maximization has been applied as generating an image that maximizes the final activation score [18]. Then, Yosinski et al. introduce a tool using the similar method to visualize each layer of CNN as it processes images or video [20]. Later, the work of Mahendran and Vedaldi [10] introduce the idea of natural pre-image, which restricts reconstructed images to be somewhat like natural images by adding regularization terms. Also, they propose a common framework of energy minimization to solve inversion, activation maximization and caricaturization problems as ways of visualizing CNN. Despite their works focusing on different issues related visualization by Inception, the common algorithm used in their researches is the gradient-based image reconstruction.

Gradient-based image reconstruction, sometimes or gradient-descent approach, is a way to heuristically modify input image to minimize the objective energy function. The objective functions in most cases contain a loss function and regularization terms. In Mahendran and Vedaldi work [10], the objective function is expressed as

\[ f(x) = R_\alpha(x) + R_{TV^\beta}(x) + CL(\phi(x), \phi_0) \]  

(2.10)

where \( R_\alpha \) is the regularization term called bounded range, \( R_{TV^\beta} \) is the regularization term called total variation, \( l \) is the loss function comparing reconstructed image \( \phi(x) \) to target image \( \phi_0 \), \( C \) is a trade off term between loss function and regularizers. Noted here, in Google’s Inceptionism [12], jitter is used as regularizer instead of bounded range and total variation. Jitter randomly shift the input image before forwarding to the network in each iteration, Equation 2.11,

\[ jitter(I,j)(x,y) = I(x + j_x, y + j_y) \]  

(2.11)

where \( x + j_x \) and \( y + j_y \) should be within image’s boundary. This regularizer interpolates missing pixels resulted by down-sampling in the bottom layers [10]. So after applying jitter, the reconstructed image will have a larger and more complete pattern of desired category.
2.3.2 Our Modification

My implementation of Inception approach is based on the gradient-based method proposed at Google’s Inceptionism [12] with additional features, such as using jitter as regularization term, normalizing the image data after each iteration and applying Gaussian filter as a way of smoothing the image. The pseudocode is given in Algorithm 1. Jitter is a crucial factor in the algorithm to make the visualization more recognizable and natural looking with a complete pattern, showed on Figure 2.3,

which compares the visualization of tabby cat category from ImageNet category with or without jitter. It is intuitive that jitter play a significant role in preserving the whole structure of category to be visualized.

Applying the above Inception implementation directly to our fine-tuned network is hard. The networks are fine tuned from the 1000 category AlexNet model pre-trained with ImageNet data. Since fine tuning is a way of tweaking existing model’s parameters and reorganizing them to make a prediction for the new classification task, the Inception visualization of each category in the new model will resemble one of the category in the pre-trained model. Figure 2.4, the visualization of April category for the AMOS data set-Golf Course, resembles the pre-trained category flamingo from ImageNet in some extent, but also capture the patterns
like the "tree branches" and the color of flowers in the fine tuning model. The results are more indistinguishable for the wheat heading task visualization on the model fine tuned from ImageNet model, Figure 2.5 demonstrates the visualization for 100 percent wheat heading from the fine tuned classification model. This could happen, since the CNN can be fooled by images of totally different content generated by gradient ascent approach which continuously optimize the final activation score [13]. The reconstructed images can be classified as right category with 99.99% confidence but still unrecognizable for human vision [13]. However, for our case, the goal is to make the reconstructed images recognizable. One intuition here is to make the parameters in CNN capture more information of new categories in the fine tuning model. So the fine tuned CNN model should learn more details and store more features of the new data set, rather than slightly tweaking parameters to achieve high activation score and make correct classification. My attempt is to make new categories in the fine tuning model more distinctive. In order to make the new categories more distinctive than the categories already existed in the pre-training model, I have tried experiments with different training policy and combination of categories for wheat heading. The detailed information of models and training policy will be discussed in the Section 3.2.

For the visualization implementation used in the experiment part, I choose jitter as regularization term and apply Gaussian filter to make the reconstructed image more smooth.
In Section 3.2.3, I use a series of experiments to show in how the visualization works for each model and what training policy could make a good Inception visualization.
Figure 2.4: The visualization for April resembles the flamingo in the pre-trained model
Figure 2.5: The visualization for 100 percent wheat heading
Chapter 3

Experimental Results and Applications

In this section, I will show the results of above approaches by applying them to two classification tasks, Visualizing Time of Year and Visualizing Wheat Heading. For Visualizing Time of Year, the results of Image masking and Taylor decomposition approaches will be showed for the five webcams. For Visualizing Wheat Heading, the results of image masking, Taylor decomposition and Inception approaches will be given for the images which have different heading percentage. We will conduct several experiments for different type of models, including regression model and classification model. The CNN training platform I am using through the experiment is the Caffe [7], a deep learning framework which achieves high speed for training and provides interface to perform fine tuning. Also Caffe has a decent binding with Numpy libraries, which makes it easy for analyzing data and do some computations in the intermediate layers.

3.1 Visualizing Time of Year

The scene classification with time of year is one of CNN fine tuning task that I have been working with. To get the fine tuned models, we start with the AlexNet model pre-trained on ImageNet. Then the network structures are modified to have 12 outputs corresponds to 12 month in the last layer instead of the original 1000 outputs. And all the intermediate layers are retrained by setting a certain learning rate. The pre-trained models are fine tuned with example images with label of which month they are taken. In this task, the webcams are
fixed and the mainly difference among the categories is the seasoning change. Sometimes, the task is so challenge that the scene captured by webcams changes little for two adjacent months. So it is crucial to have a vivid visualization of which features are important factors for making final classification decisions. In this experiment, the goal is to visualize CNN models for classifying images according to the time when this image is taken. The images all come from AMOS, which is a collection of long-term time-lapse imagery from publicly accessible outdoor webcams around the world [6]. In order to demonstrate these visualization approaches are feasible for most of the cases, five webcams are selected with different outdoor scenes. Some of the webcams locate at the geographic locations where have four distinct seasons, but others locate at the places that do not have distinct changes during the year. Hence, the CNN visualization can find out the changes which are unnoticeable for human eyes. In the following, the results of visualization by image masking and visualization by Taylor decomposition will be showed for each AMOS model.

The first webcam we chosen here is a view of golf course, which contains objects like deciduous trees, grassland, and pond, with distinct changes during the year. The second webcam is a view of Mediterranean island to the south of Greece, which has a Mediterranean climate meaning dry summers and mild to cool, wet winters. The third webcam is a monitoring camera of Superior National Forest, which locates at an area with cold and long winter. The fourth webcam is from Sequoia National Park in California, which captures the view of Alta Peak. The last one comes from the well-known scenic spot, Half Dome in the Yosemite National Park.

### 3.1.1 Visualization by Image Masking

In this part of the experiment, we select the fifth convolutional layer for visualization, and the top 3 units from total 256 units are chosen to apply mask to get significant region. Figure 3.1, 3.3, 3.5, 3.7, 3.9 are Image Calenders for the most representative images of each month, which means these images are those resulting highest final activation score in each category. Figure 3.2, 3.4, 3.6, 3.8, 3.10 demonstrate the region of image with significant
influence on the final activation score.

For the first scene, the tall tree on the left, two small trees on the bottom right and also the forest appear most of the time as significant objects resulting in high activation. The reason could be these deciduous trees keep changing during the whole year, which can be observed from Figure 3.1. The grassland seems like also contributing for the final activation score in some extent, which is reasonable since the grassland is covered by snow during the winter.

The second scene changes little during the whole year, except the fog in the fall. The most representative images for August, September and November are all the blurred scene with fog, which indicates the fog is usual during the fall. Figure 3.4 shows that mountain, village and sky (may be cloud) are the significant region for the images taken in the winter, spring or summer. But for the April, the stick showing in front of the camera is also considered as the most significant region, which means the stick appears in most of the images taken in the April and considered as a factor by CNN to classify those images to be in the April category.

The third scene comes from the Superior National Park with the view of lake, trees and grassland. The image calendar indicates that this region has a long winter with snow covering for five months. For images taken in January, the significant region is the icebound lake, the tree on the right and also the deer. For the April, the region become the lake, grassland and the trees. Trees are the dominate factors for images of scene taken in the July, which can be observed from the image that the tree on the left grows leaves only on July compared with other three months. For the October, the significant factors becomes mainly the lake and grassland, since the lake has reflection of sunlight which can not be observed in other months.

The fourth webcam locates at Sequoia National Park in California, where the vegetation is the mix of evergreen and deciduous. Actually, it is hard for human to recognize which image belong to which month. But the image masking visualization could in some extent give us hints by accentuating the significant region. The visualizations show that the mountains and the trees in the front of camera have great influence for the final classification result. After carefully observation, we find that the deciduous tree locating at the left bottom of the image gradually falls leaves during the year. This is a subtle change and may be ignored.
by human inspection.

The fifth scene is the well-known site Half Dome in Yosemite National Park. The image masking visualization shows that the cliff, forest and grassland are the significant objects. Their changes during the year can be easily observed by human eye, such as the cliff are covered by snow during the winter and the grassland gradually turns from green to brown. However, the visualizations also suggest that the sky and the tag in the top left corner have significant contribution for the final activation score.

3.1.2 Visualization by Taylor Decomposition

In order to have an intuitive sense of high relevance region, the heat maps are generated as adding red color with different intensity to the desired region on the gray-level images. So the region without red indicates there is no obvious relevance with the final activation score, since we set the pixels which are pooled out during the forward pass to be zero relevance when back-propagating through the pooling level. The intensity of red indicates the value of relevance, Figure 3.11, 3.12, 3.13, 3.14, 3.15 showing results of all month categories from the five models.

The heat maps for the first webcam have sensitive region marked as the tall tree on the left for most of the month and these two short trees on the bottom right for January, February, April, and October. The high relevance region match the high activation region marked out by image masking approach for some month, such as January and October. We learn from the results that the trees have more change than other objects in the scene and thus have a high impact on the final classification decisions.

For the second scene, the mountain, village and the stick coming in front of the camera are marked as high relevance region. But for the August and September, the visualizations do not give high intensity to any region, except some corners indicating that there is no obvious relevance between the objects in the scene and the final activation.
The heat maps of third scene show some amazing results. For January, the dear comes out as the high relevance object, which means that the dear standing on the icebound lake makes the most of contribution to classify this image as January. Also, during the icebound season, the lake has high relevance value comparing with other seasons, when the trees have high relevance value. These unnoticeable factors can be captured by CNN, but they are sometimes difficult for human to capture. And it is hard for human to tell which month each of these images belong to.

For the fourth webcam, the heat maps demonstrate that most of region marked as high relevance are the same as that marked as high activation region by image masking. The mountains in the background and the trees in the front of camera are the objects that have high relevance value. And the interesting thing happening in this visualization is that the tag on the top left also marked as high relevance region. The possible reason could be that the tag has time stamp on it and the change of time stamp has been captured by CNN.

The heat maps of the scene from the fifth camera gives a reasonable explanation by showing region that accord with our understanding of seasoning change. The mountain top, the grassland on the front, and the cliff on the left are marked as high relevance region. By manually visual inspection, we could find that snow cover actually changes during the year, and the grassland turns from covered by snow to green and eventually to brown. These visualization results show that CNN capturing features that can reasonably account for why this image belong to that particular category.

3.2 Visualizing Wheat Heading

Another experiment is to visualize CNN model for wheat heading. The wheat heading CNN models are also sharing the same 8-layer architecture as ImageNet (AlexNet architecture). But in this experiment, I visualize several models with different number of outputs and training policy to explore how these visualization approaches work on different type of models.
Table 3.1: Visualization approaches used for each model

<table>
<thead>
<tr>
<th>Models</th>
<th>Image Masking</th>
<th>Taylor Decomposition</th>
<th>Inception</th>
</tr>
</thead>
<tbody>
<tr>
<td>regression model, trained from scratch</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>classification model, 2 category, trained from scratch</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>classification model, 2 category, fine-tuned from AlexNet</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>classification model, 219 category, fine-tuned from AlexNet</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>classification model, 1002 category, fine-tuned from AlexNet</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

One type of models is the classification model with certain number of outputs, which divides the heading percentage into distinct categories. The regression model, instead, only has one output indicating the wheat heading percentage directly. Table 3.1 summarize all the models used in this experiment and what visualization approaches are applied to them. They are regression model trained from scratch, classification model trained from scratch with two outputs, 0 heading and 100 heading, classification model fine tuned from ImageNet with two outputs, 0 heading and 100 heading, classification model fine tuned from ImageNet with 219 outputs, 100 categories from ImageNet that are closest to the wheat heading, 117 random categories from ImageNet and 2 wheat heading categories 0, 100, classification model fine tuned from ImageNet with 1000 categories from ImageNet and 2 wheat heading categories, 0, 100. For the regression model, we will apply image masking and Taylor decomposition as visualization approaches to analyze the significant factors affecting final classification results. Then we also choose four classification models with different number of outputs and training policy to apply all these visualization approaches. For the classification models with 2 outputs that are trained from scratch, we will do all the experiments of image masking, Taylor decomposition, Inception. The results from image masking and Taylor decomposition will be compared with those of regression model. Then the results of visualization by inception will be compared with those generated by the rest of classification models to explore the best training policy for Inception.
3.2.1 Visualization by Image Masking

In the first part of the experiment, I train the model with 2 outputs from scratch. The reason why choosing just two categories instead of the whole categories ranging from 0 to 100 is that those two categories have distinct difference between each other, such as containing headings everywhere verse do not contain any heading. Other intermediate heading categories are transitions between 0 heading and 100 heading with different heading density. From the input images, we can see that 0 heading images entirely contain wheat leaves, but 100 heading images are composed of headings with high density and also some leaves. The visualizations by image masking are used to confirm whether those headings cause high activation, if not, what region or objects should account for the main influence for the final activation score.

Figure 3.16, 3.17, 3.18, demonstrate results of visualizing regression model. The high activation region seems to locate randomly for 0 heading visualization. For 50 heading, the results accentuate some region which contains wheat headings growing upward. Finally, the results of 100 heading also suggest that the headings are the features resulting high activation. From the visualizations, we could see that the leaves and headings are the features resulting high activations, which is a reasonable explanation, since the main difference among these categories is the density of headings. Figure 3.19, 3.20 are the visualizations from classification model with 2 outputs, which shows the similar results but with the high activation features from different position in the image. This is probably the drawback of using image masking to visualize such kind of CNN model like wheat heading, where the objects or features in the same category oriented in different ways. Also, there are repeating objects, such as leaves and headings, of the approximately same activation score. It is hard to decide to which one has the most significant influence on the final activation score.
3.2.2 Visualization by Taylor Decomposition

We apply visualization by Taylor Decomposition to both the regression model and classification model, which are the same as what we used for image masking. Figure 3.21 shows the result of regression model for 50 heading on the left and 100 heading on the right, the reason why we select 50 heading instead of 0 heading here is that the final activation score for 0 heading is 0 in the regression model. So the back-propagation will generate all 0 relevance for the input pixels. The visualizations of regression model give us a hint about the regression CNN mainly focusing on the region around the corners. In the visualization of 50 heading, the ground on the bottom left comes out as the high relevance region. Probably, for most images classified as 50 heading, the bottom left region is different from that of images from other categories. In the visualization of 100 heading, the headings locating along the margins, are marked as red, so these headings account for why the image belong to the category of 100 heading. The possible reason could be 100 heading images have more headings appearing along the margins compared with the other heading percentage, such as 50, 70, 90. The visualization results of classification model with 2 outputs are interpretable, Figure 3.22. The high relevance pixels are mainly those of the leaves in the visualization of 0 heading, and most of heading are marked as high intensity on red for the case of 100 heading. From the visualization results of classification model, we can conclude that leaves and headings are the features that have high relevance with the final activation score. Visualization by Taylor decomposition works better than image masking for the task of visualizing model of predicting wheat heading, which shows the relevance for all the pixels instead of finding those which maximally activate the final score. Hence, the Visualization by Taylor decomposition is more suitable for the case where multiple features in a certain image are highly related to the final classification decision.

3.2.3 Visualization by Inception

I select four classification models with different training policy to explore how Inception approach works for each case. For the purpose of comparison, all the model are trained
around 50 epochs, and go through same number of iterations during the gradient descent reconstruction process. One epoch means exactly one forward pass and backward pass of all the training examples. Figure 3.24 is the results of visualizing categories of 0 heading and 100 heading. We can see that the CNN model store the information such as color and the shape of leaf. The main difference between these two visualizations is the color. Probably for this training from scratch model, the main factor for CNN to make classification is the color. Probably for this training from scratch model, the main factor for CNN to make classification is the color. Figure 3.25 and Figure 3.26 both demonstrate that the results from the fine tuned model contain the ”wave”, which might come from the shape of the leaf in the wheat images. Comparing with the results from the training from scratch model, we find that fine tuning CNN mainly capture the shapes and patterns as important information instead of color. Since the weights and bias in the fine tuning model are tweaked from the original ImageNet model, which have one thousand categories, the new model may still contain the color information stored by the previous model. Another possible reason could be that a certain wheat heading category contains images with headings randomly spread and orient, so there is not a certain pattern with regularity for this category. If we compare the visualizations generated by CNN for wheat heading with those of tabby cat in ImageNet, Figure 2.3, we could find that the Inception method generates better visualization for the original training from scratch ImageNet model than these from wheat heading.
Figure 3.1: Most representative image for each month, month ordering from left to right, top to bottom
Figure 3.2: Visualization results for AMOS: the golf course. January, April, July and October are selected as a representative month for each season. The row represents month and the column represents unit.
Figure 3.3: Most representative image for each month, month ordering from left to right, top to bottom
Figure 3.4: Visualization results for AMOS: Mediterranean island. January, April, July and October are selected as a representative month for each season. The row represents month and the column represents unit.
Figure 3.5: Most representative image for each month, month ordering from left to right, top to bottom
Figure 3.6: Visualization results for AMOS: Superior National Forest. January, April, July and October are selected as a representative month for each season. The row represents month and the column represents unit.
Figure 3.7: Most representative image for each month, month ordering from left to right, top to bottom
Figure 3.8: Visualization results for AMOS: Sequoia National Park. January, April, July and October are selected as a representative month for each season. The row represents month and the column represents unit.
Figure 3.9: Most representative image for each month, month ordering from left to right, top to bottom
Figure 3.10: Visualization results for AMOS: Half Dome, Yosemite. January, April, July and October are selected as a representative month for each season. The row represents month and the column represents unit.
Figure 3.11: Heat map for the scene from golf course
Figure 3.12: Heat map for the scene from Mediterranean island
Figure 3.13: Heat map for the scene from Superior National Forest
Figure 3.14: Heat map for the scene from Sequoia National Park
Figure 3.15: Heat map for the scene from Half Dome, Yosemite
Figure 3.16: Visualization results by image masking for 0 wheat heading on regression model
Figure 3.17: Visualization results by image masking for 50 wheat heading on regression model
Figure 3.18: Visualization results by image masking for 100 wheat heading on regression model
Figure 3.19: Visualization results by image masking for 0 wheat heading on classification model with 2 outputs
Figure 3.20: Visualization results by image masking for 100 wheat heading on classification model with 2 outputs
Figure 3.21: Visualization results by Taylor decomposition showed as heat map for regression model, left: 50 heading, right: 100 heading
Figure 3.22: Visualization results by Taylor decomposition showed as heat map for classification model, left: 0 heading, right: 100 heading
Figure 3.23: Visualization results by Inception for the classification model trained from scratch with 2 outputs, 0 heading on the left and 100 heading on the right

Figure 3.24: Visualization results by Inception for the classification model fine tuned from ImageNet with 2 outputs, 0 heading on the left and 100 heading on the right
Figure 3.25: Visualization results by Inception for the classification model fine tuned from ImageNet with 219 outputs, 0 heading on the left and 100 heading on the right.

Figure 3.26: Visualization results by Inception for the classification model fine tuned from ImageNet with 1002 outputs, 0 heading on the left and 100 heading on the right.
Chapter 4

Conclusion

A Convolutional Neural Network achieves state-of-art performance in large scale image classification and pattern recognition tasks. Much research sheds light on improving their accuracy, but it is also important to develop an intuition to understand which factors play important roles in CNNs classification decision making. The visualization approaches act as tools to make the network transparent and explain how these networks operate to classify images by showing visual results generated base on the data stored in the CNN. In this thesis, I study three visualization approaches, image masking, Taylor decomposition back-propagation and Inception, from different aspects and apply them to two image classification tasks, Visualizing Time of Year and Visualizing Wheat Heading. I either implement or modify these approaches to apply to our cases, the visualizations generated from different approaches are compared to shed lights on which factors are treated as significant influence for final classification results.

I apply the visualization approaches to the uncommon image classification tasks, where the adjacent categories have small and unnoticeable difference or the images in the same category have patterns randomly oriented. These categories are sometimes hard for human to distinguish. So there should be a way to understand how CNN work to solve classification task like that. To achieve this, I use image masking, Taylor decomposition back-propagation and Inception as visualization approaches to interpret which factors have significant contribution for the classification decisions. From the experimental results, we conclude that the visualization approaches can give reasonable explanation to these classification tasks about why this image belong to that particular category.
In the thesis study, I explore how these visualization approaches work on the regression model, which is a novel application. Usually, researchers working on visualizing CNN focus on more intuitive interpretation for classification model, few of them do the experiments for regression model. Here, I apply these visualization approaches to regression model and also get reasonable explanations.

Three visualization approaches are compared by the results from the experiments. The image masking and Taylor Decomposition perform well for fine-tuned models, both classification model and regression model. Image masking approach aims at finding the region with the most significant contribution to the final activation score, but the Taylor Decomposition back-propagation approach interprets the classification decisions by the pixels which have relevance to the final activation score. The visualization by inception works well for the training-from-scratch models, but it cannot generate an intuitive result based on information stored in the fine-tuning model.

Considering the case that visualizing by Inception cannot generate a visualization showing complete object structure for fine tuning model, we could explore approaches to visualize CNN by marking objects with more complete structure in the future research, such as marking the whole tree as significant object in the visualization for the golf-course model. We could investigate a way to segment scene images into several objects or images of an object into semantic meaningful parts. For example, we segment the scene images into trees, buildings, vehicles and so on or the images of the cat into eyes, ears, legs and so on. Then a visualization would demonstrate the relevance of these objects or parts to the final classification decision. So we can obtain a more intuitive understanding of the CNN classification by the objects contained in the image.
References


Vita

Dingwen Li

Degrees

B.S. Electrical and Computer Engineering, Shanghai Jiao Tong University, July 2014
M.S. Computer Science, Washington University in St. Louis, May 2016

May 2016
Visualization of Deep Convolutional Neural Networks, Li, M.S. 2016