COMP 569 Final Project
Image Recognition on Busy Images with Convolutional Neural Networks

Mark I. Edwards

December 14, 2016

Abstract

Convolutional Neural Networks (CNNs) are one of the most up-and coming image recognition algorithm. In this project, I aim to use CNNs to perform image recognition on busy images, locating small targets in images containing many background patterns. I do this by replicating the work in [1] and use and use CNNs to perform image recognition on aerial images of vehicles.

Introduction

Image recognition on busy images can be thought of as the task of teaching computers to play Where's Waldo.™ It is the task of teaching computers to identify faces within a crowd, to identify animals within a landscape, or to identify cars, buildings and roads in an aerial image. In this project, the author examines the task of identifying cars within aerial images, using the Vehicle Detection in Aerial Imagery (VEDAI) dataset [6].

Background

Prior to the rise of Convolutional Neural Networks, a number of other approaches were used to perform image recognition on busy images. One such method was known as template matching. Some templates extract features, such as edges, curves or points, and others consist of small pixel areas that are compared to patches within the image containing the target to be matched with some similarity measure [5]. Such rudimentary learning methods and other expert systems were popular through the 1960s and 1070s [4].

Starting in the 1980s, simple machine learning classifiers gained popularity [4]. One of the most popular machine learning algorithms used for analyzing aerial images was the Naive Bayes classifier. Early approaches attempted to classify every pixel i into a class $c+i$ in the input image based on local features x_i. The primary drawback of this is that it requires that

we specify the class-conditional distribution $p(x_i \mid c_i = k)$ for each class k. The multivariate normal distribution is typically used, but this limits the classifier to only linear or quadratic decision boundaries. Neural Networks rose as an alternative to Bayesian models as they can approximate the class-conditional distribution as a differentiable function with learned parameters, both eliminating the restriction on decision boundaries and the requirement that the class-conditional distribution be explicitly found.

Another method that gained traction was the use of machine learning "ensembles." Machine learning ensembles combine multiple machine learning models, typically either through Boosting or Averaging. Boosting is the process of using multiple weak machine learning algorithms in succession, to gradually refine results. A common boosting algorithm is the AdaBoost "Adaptive Boosting" method, which integrates multiple Tree learning algorithms in succession. AdaBoost is itself a stage within one of the best known image recognition algorithms, the Viola-Jones algorithm. An averaging algorithm instead uses multiple weak classifiers in parallel, averaging the output. One example of a commonly used averaging algorithm is the Random Forest method. In the Random Forest method, multiple decision trees are trained on randomly selected subsets of the training data taken with replacement using a random subset of features in a process known as "Bagging." The predictions of the different trees are then averaged to get the final result. The random forest algorithm has shown considerable success at detecting brain tumors within MRI scans [3].

To improve further upon the successes of ensemble methods the authors of [1] have turned to Neural Networks. Deep Neural Networks have a structure that effectively performs boosting and averaging on neurons following the perceptron algorithm. In particular, Convolutional Neural Networks (CNNs) have shown considerable prowess at image detection.

Convolutional Neural Networks

Neural Networks are a paradigm inspired by the function of the biological brain. Artificial Neurons simulate the function of biological neurons using a weighted sum of their inputs and an activation function f as shown in Figure 1 and Figure 2.

However, individual neurons are only capable of categorization along linear boundaries. In order to achieve the non-linearity needed for more accurate predictions, multiple neurons must be connected into a network as shown in Figure 3. In fact, a sufficiently large Neural Network can approximate any continuous function on compact subsets of $\mathbb{R}^n$ [2].

However, fully connected Deep Neural Networks are computationally inefficient and prone to overfitting. This is because in a fully connected neural network, each neuron in the first hidden layer (the layer after the input layer) receives information about each pixel in the image, which is often computatationally untenable and extraneous. Convolutional Neural Networks employ "spatial sparsity" in the form of convolutional layers.

In the field of Image Processing, the word convolution or kernel refers to a linear operation that takes place over a small region within a larger

Figure 1: Biological and Artificial Neurons

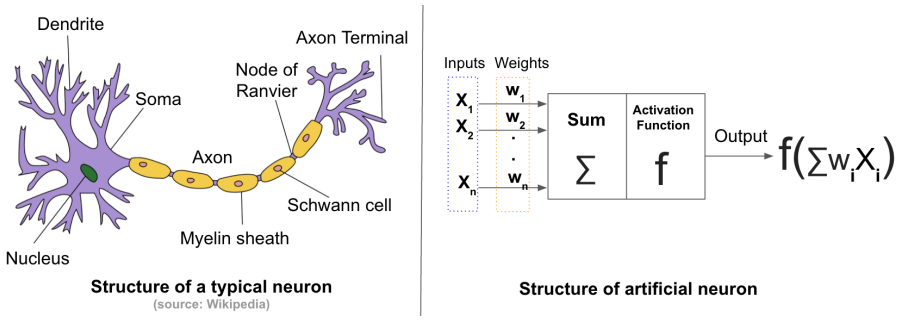


Figure 2: Activation Functions

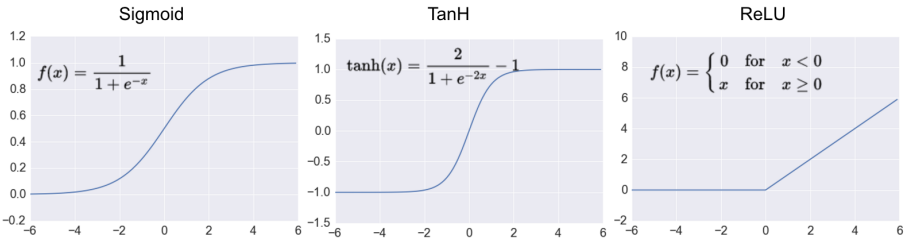


image as shown in Figure 4. In a convolutional neural network, such convolutions take the form of connections within the network corresponding to a region. In essence, the convolutions are learned rather than predetermined.

Typically, CNNs employ multiple learned convolutions in their convolutional layers. This results in much more data than the original input, which adds to memory and processing costs. Since some of this data is extraneous, a form of sub-sampling called "pooling" is often employed. One popular form of pooling is "max pooling." In max pooling, many small spatial regions of the output of multiple convolutions are selected. From these regions, only the largest value is kept. This reduces computational cost and reduces overfitting, without much loss of information.

The exact CNN used in this project is shown in Figure 5. It was developed in [1] with inspiration from [4]. It employs a large initial convolution that allows it to closely match small targets. It was originally applied to the Massachusetts Roads Dataset and the Massachusetts Buildings Dataset. However, the author of this project was unable to access those databases, and instead used the Vehicle Detection in Aerial Imagery (VEDAI) dataset [6]. In addition, many training methods implemented within [1] were not implemented, such as model averaging over spatial displacement, to account for small changes in target position, and data augmentation by randomly rotating the input images. As implemented, this CNN takes a 512×512 3 color image as input, and outputs a 16×16

Figure 3: Neural Network

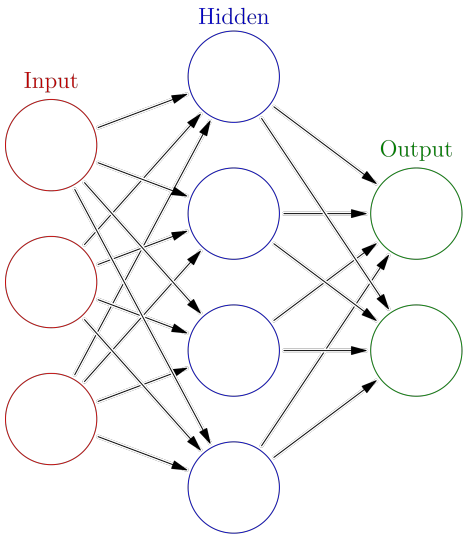


Figure 4: Convolution

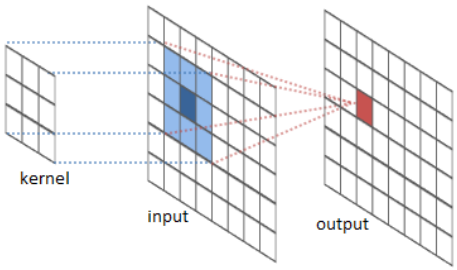
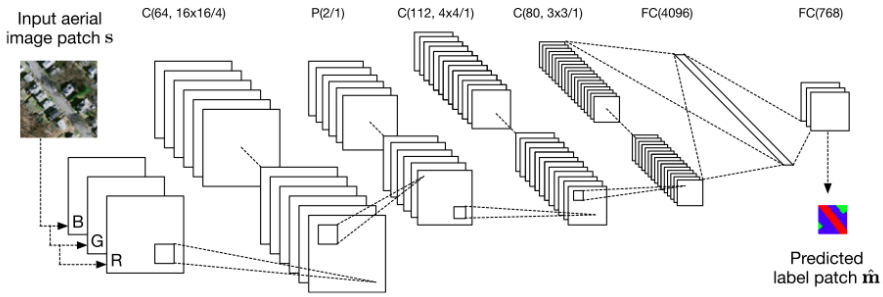


Figure 5: Minh Net



grid with a pixel depth equal to one more than the number of classifica-

tions (an additional background classification).

References

[1] Multiple object extraction from aerial imagery with convolutional neural networks. *Journal of Imaging Science and Technology* (2016).

[2] Hornik, K. Approximation capabilities of multilayer feedforward networks. *Neural Networks*.

[3] Menze, B. H., Jakab, A., Bauer, S., et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging 34*, 10 (Oct 2015), 1993–2024.

[4] Mnih, V. *Machine Learning for Aerial Image Labeling*. PhD thesis, 2013.

[5] Perveen, N., Kumar, D., and Bhardwaj, I. An overview on template matching methodologies and its applications. *International Journal of Research in Computer and Communication Technology* (2013).

[6] Razakarivony, S., and Jurie, F. Vehicle detection in aerial imagery : A small target detection benchmark. *Journal of Visual Communication and Image Representation, Elsevier* (2015).