Fundamentals of Target Classification Using Deep Learning

Irene L. Tanner[a], Abhijit Mahalanobis[a]
[a]University of Central Florida, 4000 Central Florida Blvd, Orlando, FL, USA 32816

ABSTRACT

In this paper we examine the application of deep learning for automated target recognition (ATR) using a shallow convolutional neural network (CNN) and infrared images from a public domain data provided by US Army Night Vision Laboratories. This study is motivated by the need for high detection and a low false alarm rate when searching for targets in sensor imagery. The goal of this study was to determine a range of optimal thresholds at which to classify an image as a target using a CNN, and an upper bound of the number of training images required for optimal performance. We used a Difference of Gaussian (DoG) kernel to localize targets by detecting the brightest patches in an image and using these patches as testing data for our network. Our CNN was successful in distinguishing between targets and clutter, and results found by our approach were favorably comparable to ground truth.

Keywords: ATR, deep learning, classification, convolutional neural network, localization

1. INTRODUCTION

Target detection and recognition are often required for defense, surveillance, and situational awareness, yet the use of deep learning for ATR has not yet been extensively researched. There are some previous investigations on EO/IR data [1] and using the SAR MSTAR data set [2]. Infrared images can be used to train deep learning algorithms to detect military targets in natural scenes at any time of day. The Automated Target Recognition Algorithm Development Image Database provided by the U.S. Army Night Vision Laboratory contains such images [3], where there is a total of nine target classes, ranging from pick-up trucks to tanks and wheeled military vehicles (see Figure 1). Images used were taken from 1000, 1500, and 2000 meters away from several aspect angles during both daytime and nighttime. Previous research in ATR focuses on using quadratic correlation filters [4], support vector machines [5], and autoencoders [2], but the use of deep learning approaches is more recent.

Figure 1. Examples of targets *(first three images)* and clutter images *(last three images)*.

2. TECHNICAL APPROACH

2.1 Architecture

Infrared images were cropped to 80x80 pixel patches of either clutter or a target centered on the ground truth pixel location of the target. These patches were used as input to a five-layer convolutional neural network (CNN) and the output was a two-class probability distribution, with target being the positive class. The architecture of the CNN remained unchanged for most of this work (see Figure 2). Our model was built in Python using a Keras sequential model with a Tensorflow backend. In each convolutional layer, filter size is 3 x 3. The first three activation functions used are ReLU and the last is softmax. Dropout occurs after the max pooling layer and after the first dense layer.

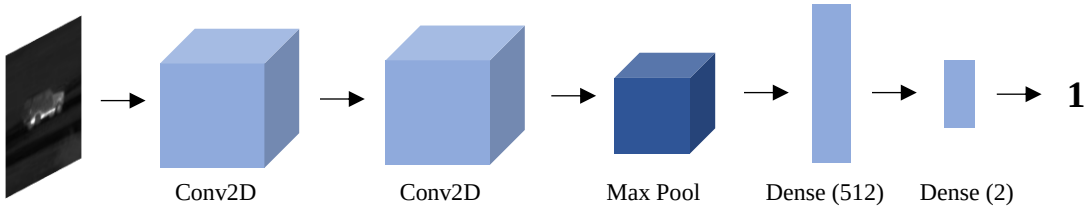

Figure 2. Convolutional neural network architecture.

2.2 Method

Initially, we used a total of 5000 images to determine if it was possible to distinguish between targets and clutter using a shallow network. Of these images, 4500 consisted of the ten target classes and 500 consisted of random clutter; 70% of this data was used for training, 20% for validation, and 10% for testing.

Once the classifier was trained to separate targets from clutter, a threshold was applied to the output score to determine whether an image is considered a target or not. The threshold was varied to find the value that produced the highest true positive rate (TPR) and the lowest false positive rate (FPR). For statistical analysis, a confusion matrix was used to compare the FPR and TPR across varied thresholds. The number of target and clutter images used to train the network was also altered and we then determined the fewest number of images that produced the highest TPR and lowest FPR. Here, ROC curves were implemented to compare these rates across all thresholds. Finally, we used a Difference of Gaussian (DoG) kernel to detect the brightest part of an image scene (here, referred to as a patch), and classified these patches using our CNN for classification. Again, ROC curves were utilized to compare the FPR and TPR when varying the types of images used.

3. RESULTS

We were able to distinguish between clutter and targets with a high true positive rate and low false positive rate (see Figure 3). Approximately 96% of targets were correctly classified as targets with as few as 4% of clutter images being incorrectly classified as targets. For reference, in each of the following ROC curves, the red line represents the threshold, the black line represents the baseline, and the blue line is the actual result.

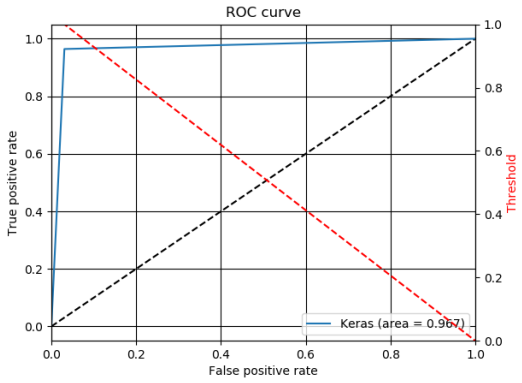

Figure 3. Initial ROC curve.

We found that both the FPR and TPR decrease as the threshold increases. Specifically, when the threshold was 0.1, all targets were correctly classified but there was a high false alarm ratio, whereas when the threshold was 0.9, fewer targets were correctly classified but there were no false positives (see Figure 4). In each confusion matrix, the upper left box

contains the TPR, the bottom left contains the FPR, the upper right false negative, and bottom right true negative. Having high values on the main diagonal indicates that each image class was correctly classified.

Figure 4. Confusion matrices at varied thresholds *(left: 0.1, center: 0.5, right: 0.9).*

Using approximately seven-, twelve-, and seventeen thousand training images produced similar successful results (see Figure 5). Each experiment had a TPR of almost 1.0 with a FPR that was very close to 0.0. It is important to note that when using 7276 images, the optimal threshold was close to 0.9, but when using more images, the optimal threshold was around 0.1.

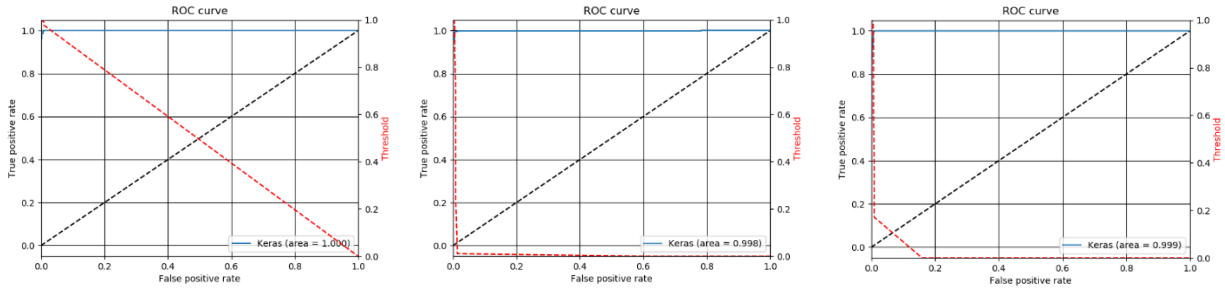

Figure 5. ROC curves for varied number of training images *(left: 7276 images, center: 12756, right: 17010).*

Large, bright targets on dark backgrounds were correctly detected using the Difference of Gaussian (DoG) method (see Figure 6). However, neither small nor dark targets were correctly detected, especially in images with high brightness and contrast (e.g. infrared images taken during daytime), and in images in which the target blended into the background. Additionally, numerous targets either had a non-centered bright spot or several bright spots scattered along the edges of the vehicle; in these images, the DoG method detected multiple targets when there was only one. When the CNN was trained with only centered targets and tested with patches found by the DoG, the ROC curve was less than optimal, as it had achieved a TPR of only 0.65 when the FPR was the lowest. However, when the network was trained with non-centered targets – targets shifted ten, twenty, and thirty pixels away from the ground truth center pixel location of the target – the ROC curve dropped below the baseline and became almost linear (see Figure 7).

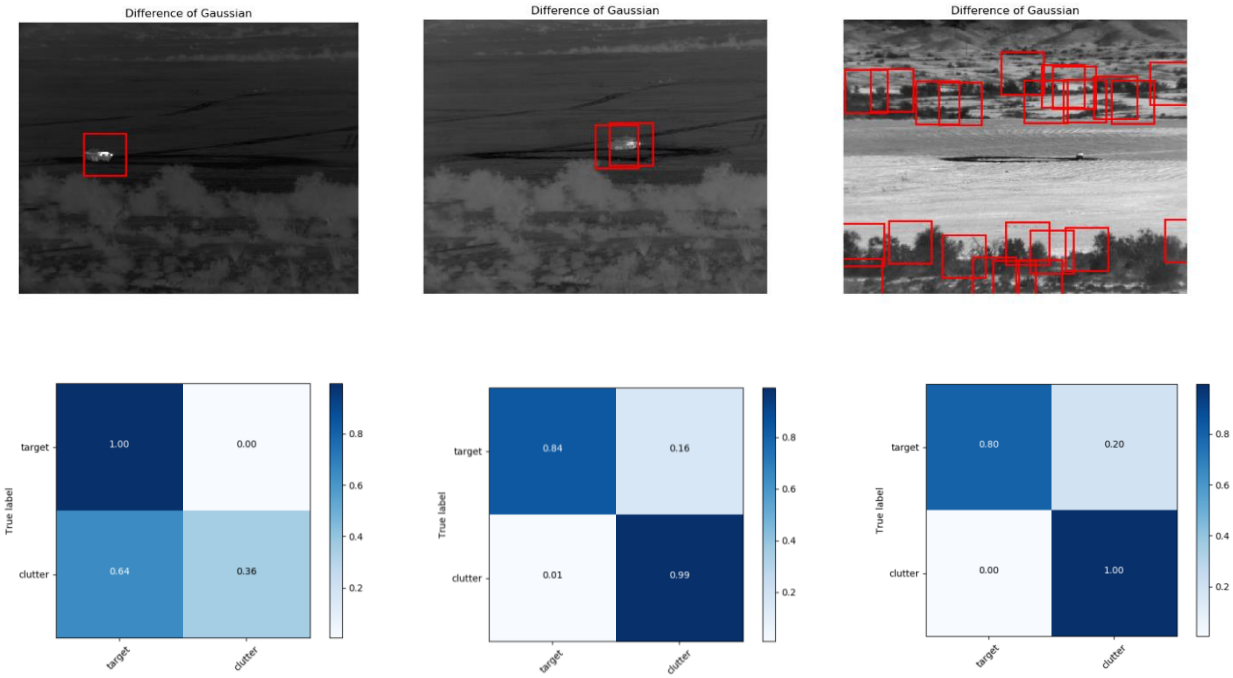

Figure 6. Examples of patches found by DoG method *(left: target was correctly detected; center: two bright spots on one vehicle were detected; right: target was not detected and there was a high FPR).*

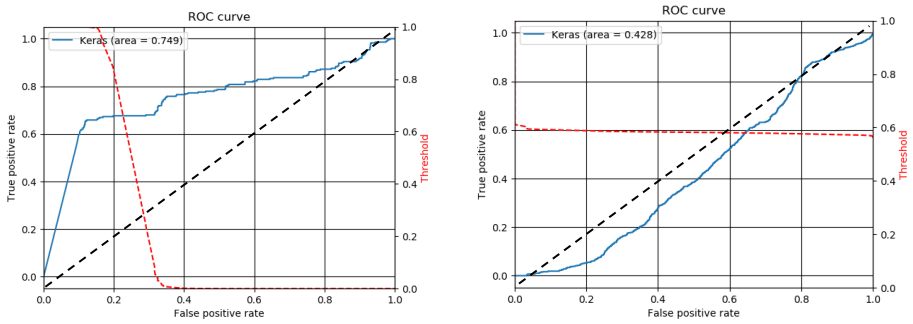

Figure 7. ROC curves found with DoG method *(left: trained with centered targets; right: trained with non-centered targets).*

Implementing a more complex CNN produced an ROC curve (see Figure 8) comparable to the one found in Figure 3. This network implemented bias regularizers as well as more filters in the first two convolutional layers. The TPR was higher and the FPR lower at each threshold than those found in the curves using both centered and non-centered targets.

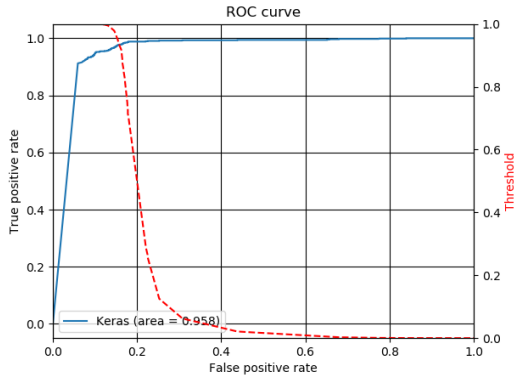

Figure 8. Improved ROC curve using a more complex CNN.

4. SUMMARY AND CONCLUSION

Based on the analysis presented in the previous section, we found that the optimal threshold for classifying an image as a target lies between 0.5 and 0.9, where 0.5 minimizes the FPR and has a relatively high TPR. We then found that using several thousands of training images produces negligible differences in the FPR and the TPR, most likely due to the redundancy of the data. Therefore, we can get similar results using a smaller dataset size (e.g. fewer than 7000 training images), thus reducing computational overhead. We also discovered that the simple DoG method is ineffective for accurately detecting targets by itself. This is because patches are assumed to be bright on a dark background, the brightest part of a target is not necessarily in the center, the FPR increases with higher average pixel intensities, and it is difficult to detect targets at long ranges. Further, we found that testing the shallow CNN with non-centered targets found by the DoG was inconclusive, as the network was too simple and shallow to process this more complex data. However, after implementing a more complex network, our results improved significantly. The ROC curve produced by this method on full scenes was comparable to the curve produced by the network that was meant solely for classifying image patches on target and clutter distinction. In the future, we want to conduct more research on finding the optimal dataset size and making the DoG+CNN method more effective for target detection. Further refinement of the classifier can be done to classify different vehicle types (e.g. military v. civilian vehicles) and then identify the exact vehicle type. Thus, classifying detected targets using deep learning appears to improve overall performance while decreasing the false alarm ratio, which is ultimately necessary to recognize and respond to threats on the battlefield.

REFERENCES

[1] Nasser M. Nasrabadi, Hadi Kazemi, Mehdi Iranmanesh, "Automatic target recognition using deep convolutional neural networks," Proc. SPIE 10648, Automatic Target Recognition XXVIII, 106480I (30 April 2018);

[2] Deng, Sheng, et al. "SAR Automatic Target Recognition Based on Euclidean Distance Restricted Autoencoder." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 7, 2017, pp. 3323-3333.

[3] Automated Target Recognition Algorithm Development Image Database. Defense Systems Information Analysis Center 1.49d (2014).

[4] Mahalanobis, Abhijit, et al. "Design And Application Of Quadratic Correlation Filters For Target Detection." *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 3, 2004, pp. 837-850.

[5] Zhao, Q., and Principe, J. C., "Support Vector Machines For SAR Automatic Target Recognition." *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 2, 2001, pp. 643-654.