Automated Region of Interest Creation for Training Sets

Heather Vooys

July 17, 2020

Preparatory material

Calculus, image processing, numerical methods as well as some knowledge of machine learning methods would be useful.

Introduction

Many machine learning techniques need training prior to making predictions about an input. These training sets often require large amounts of data to train the model well. Preparation of such data sets can take years. The ImageNet dataset took three years to build with a large team of people [1]. The dataset contains over 14 million images and only approximately 1 million of those images have annotated bounding boxes around objects of interest [2].

Depending on the problem you are facing, data is often available to face it using machine learning; however, in some cases, that data does not exist. Applying machine learning to industrial problems thus has an extra layer of trickiness. Not only are we teaching a model to perform an action, but sometimes it is necessary to create the teaching materials for it. This can be rather time consuming depending on the model. For instance, the region proposal models such as those in the R-CNN family require images as well as masks specifying the regions of interest. These masks are binary images which dictate the area of the image in which we are interested. How do we create these data sets for industry problems when a team and time may be limited?

Problem Description

Standard procedure for building training sets described above involved an individual going through each image and creating a 2D binary matrix which reflects where the region of interest is in the image. Afterwards, they would also ascribe a label to the image. For instance, if someone wanted to detect dogs in different images, they would create a training set containing images of dogs. Using a RCNN machine learning model means they would also need to create a related 2D binary matrix which describes where the dog, or dogs, are in each image for the training set. Given that this process relies solely on the person, human error is likely to occur. Perhaps a dog gets missed in an image containing multiple dogs, or one image contains a cat that is mistaken for a dog.

One way to remove the human error component is to limit the amount of work the individual does when creating a training set. If we are creating a data set because one does not already exist, pre-existing machine learning models would not be available for object detection. We need to find a different method for automating this training set creation. For RGB images, we can focus on the pixel color to extract specific objects; however, depending on the complexity of the object (color variations, etc.), this problem becomes difficult very quickly. For instance, dogs come in a variety of colors. Using the color information to isolate pixels related to dogs would be difficult; however, automating the extraction of monochromatic objects using color information is potentially achievable.

For example, say we wanted to create a RCNN model that detects different types of road lines. Figure 1 shows an image of a section of road on the left. There are two types of road markings: the yellow middle lines and the white edge lines.

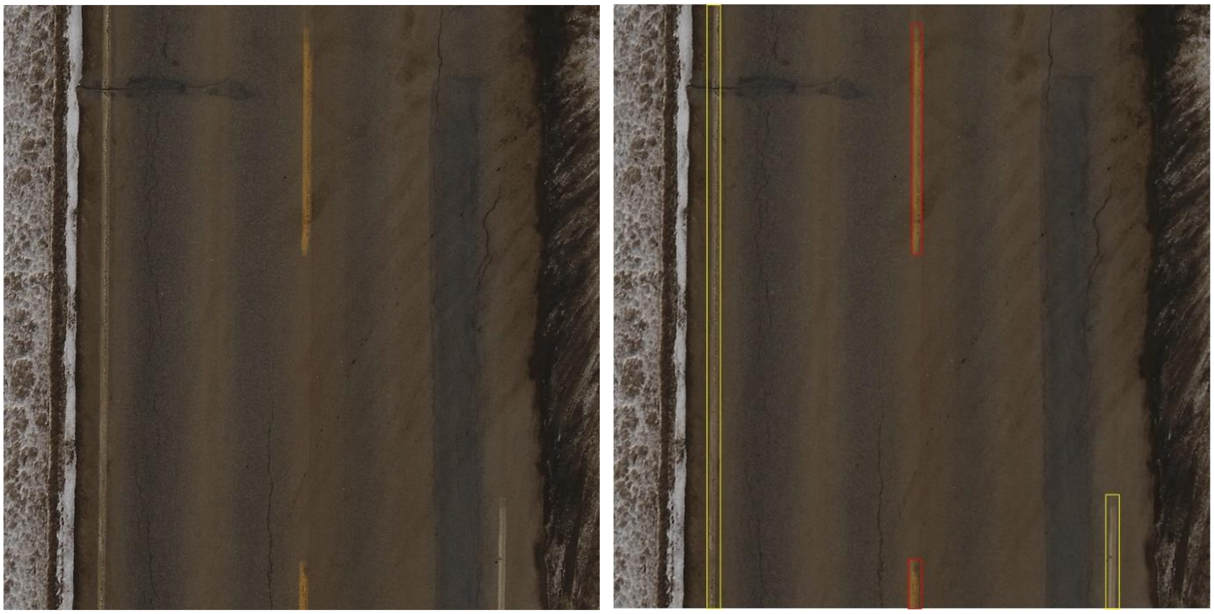

Figure 1: (Left) An aerial image of a road containing yellow middle lines and white edge lines. (Right) The aerial image of the road with red bounding boxes designating the location of yellow lines and yellow bounding boxes designating the location of white lines.

Our RCNN model needs a training set which shows where the yellow lines and white lines occur in images. Fig. 1 (right) gives bound boxes depicting where the yellow lines are (red bounding boxes) and where the white lines are (yellow bounding boxes). These boxes were handpicked by someone, but can we use RGB information or some other method to automate this process?

For this project, the team will be given access to RGB images containing different types of lines painted on a hard surface. The goal is to develop a method which creates a mask of the image depicting where the lines occur automatically and with very limited user input. There are various techniques which may help reach this goal. Given the monochromatic nature of the lines, image processing and RGB information could be useful. There are also clustering methods which can be used to cluster similar pixels; however, clusters are not always consistent between images. While these techniques are available, it will be important to find the method which gives consistent results with limited user input.

References

1. T. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár "Microsoft COCO: Common Objects in Context," in arXiv, 2014.
2. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in CVPR, 2009.