Learning to count: a Machine Learning adventure

G. Vogiatzis

Abstract

This document describes a toy inference problem that we have devised, to give you a flavour of a typical Machine Learning task. The problem is designed to stretch your hacking abilities, together with your wider analytical skills. It involves building a Machine Learning model (e.g. a deep neural network) that processes an image containing elliptical, coloured blobs and correctly counts them amongst clutter in the form of coloured polygons.

1 Introduction

Machine Learning (ML) has seen a phenomenal rise in popularity due to several related technologies that have led to breakthroughs in challenging problems. This document presents you with a challenge: Using ML tools and techniques, (e.g. deep convolutional neural networks or any other model) your solution should provide a count for a number of features present in an image. It is well known that ML techniques are data-hungry, so to keep things simple we have restricted the problem. The input to your network will be a coloured image, and the features you are meant to count will just be elliptical blobs (see Fig. 1). The problem is made more challenging by the presence of clutter, which takes the form of randomly coloured polygons.

2 The dataset

We have created a dataset for you which can be found in three different versions:

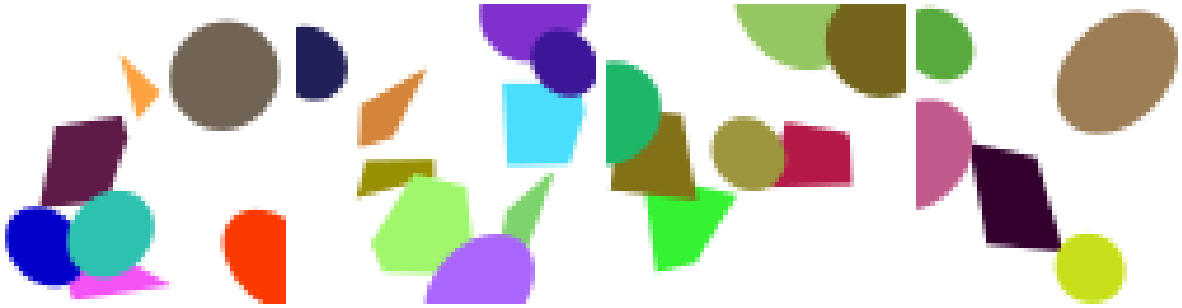

Figure 1: Four images from the shapes dataset. The images contain coloured polygons and ellipses. Your objective is to train a machine learning model to count the ellipses.

- http://george-vogiatzis.org/MScAI/shapes_dataset_LR.zip
- http://george-vogiatzis.org/MScAI/shapes_dataset_MR.zip
- http://george-vogiatzis.org/MScAI/shapes_dataset_HR.zip

The LR version contains low-resolution 64x64 images, MR version contains mid-resolution 128x128 images while the HR version contains high-resolution 224x224 images such as the ones used in the ImageNet challenge [5]. Each of the three collections consists of 10,000 RGB images, with each image containing between 0 and 5 ellipses and a number of clutter polygons of various colours. In each collection we have split the images into 6 folders (according to how many ellipses they contain) and there is also a file (`labels.csv`) that tells you the number of ellipses and number of polygons for each image. The number of polygons is included there for further investigations (e.g. does the number of clutter polygons affect the accuracy of your ellipse count etc) Figure 1 shows four images from the LR dataset, all containing four ellipses.

3 Your task

We would like you to train a Machine Learning model to predict the number of ellipses in the image while ignoring the polygons. Our suggestion is to cast the problem as classification with the six classes corresponding to 0, 1, 2, 3, 4, 5 ellipses present in an image. However you could also view this as a regression problem where your model should output a number from 0 to 6, corresponding to the number of ellipses. It's your call. Remember: in the problems we face as engineers there is rarely a single *right answer*! We leave the choice of model up to you but we would encourage you to use Python to develop your solution. If you are interested in trying out some Deep Learning technology we would recommend Pytorch [3] and in particular the tutorial on simple image classification [2]. If you would like to know some more about the theory behind Convolutional Neural Networks and Machine Learning in general, have a look at some of the material in this course from Stanford [1] as well as the excellent (and freely available) Deep Learning book by Goodfellow *et al* [4].

We have provided you with image collections of three different resolutions because, depending on your available computing power, you might only be able to run your solution on the LR or MR versions. Generally accuracy tends to improve with high-resolution images but feel free to use the resolution your computer can handle.

4 Possible directions

The task described is open ended because we wish to engage your problem solving skills and creativity without any limits imposed. Similarly to research carried out for an MSc dissertation or a PhD, this mini-investigation has no strict parameters and no single right answer. The main object of your work should be to find a Machine Learning model that works best for this problem. We are interested in challenging cases that make your algorithm fail, as well as the good cases where it works. Some other interesting points (some harder than others) to look into would be:

- What is the effect of clutter (polygons) for the accuracy of your solution?
- Can you shed some light into how your model manages to perform the computation? What type of features does it look for?
- Can you try to predict the number of polygons instead of the number of ellipses? Is this a harder problem? Why/why not?

Please prepare a brief report (using Latex or Word) that summarises your findings. In order for us to be able to discuss your solution during the interview, you should provide us with your report and your code at least one working day before the interview takes place. The code can be provided as an appendix or as a separate zip file or git repo, etc. We would encourage you to complete the task but if you only get part way through the activity, we would still be interested to know what you found. Good luck!!!

GV, April 2019

References

[1] CS231n Convolutional Neural Networks for Visual Recognition. URL: http://cs231n.github.io/.

[2] Deep Learning with PyTorch: A 60 Minute Blitz PyTorch Tutorials 1.1.0.dev20190425 documentation. URL: https://pytorch.org/tutorials/beginner/deep_learning_60min_blitz.html.

[3] PyTorch. URL: https://pytorch.org/.

[4] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. URL: https://www.deeplearningbook.org/.

[5] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 12 2015. URL: http://www.image-net.org/challenges/LSVRC/, doi:10.1007/s11263-015-0816-y.