Image Segmentation with Neural Networks for Indoor Pieces

Joël KY
Department of Computer Science and Telecommunications
Toulouse INP-ENSEEIHT
2 Rue Charles Camichel
31000 Toulouse
joelroman.ky@etu.enseeiht.fr

Abstract

Deep learning has revolutionized artificial intelligence by the introduction of neural networks. In fact, the multiple applications of neural networks allowed a significant breakthrough, particularly in image processing. The objective of this paper is to present the set up a Deep Learning technique, semantic segmentation for the recognition of the elements of an interior room and model conversion from Caffe to TensorFlow. This paper presents the different steps and choices made as well as the results obtained at the end of the implementation of the state of the art of Computer Vision architecture for Semantic segmentation.

***K*eywords** Neural networks, Model Conversion, Semantic segmentation, PSPNet, Computer Vision

1 Introduction

Deep-Learning is a type of Artificial Intelligence derived from machine learning where the machine is able to learn autonomously.
It is a promising technique, based on a network of artificial neurons inspired by the human brain. This network is made up of tens or even hundreds of layers of neurons, each receiving and interpreting information from the previous layer.
This article will present a project carried out as part of an internship within the 3D and augmented reality society, INSPI which modernizes furnishings. The objectives of the interns
hip were to

- convert a segmentation model
- collect and preprocess data used to train a segmentation model
- deploy the model on Web Services in order to be able to call and use it.

The model must be able to recognize the different elements such as the floor, the ceiling, the furniture from an indoor image.

2 Details of the project

2.1 State of the art

The first step in this project was to conduct a state-of-the-art. The first exploratory research stage consisted of reviewing the different existing platforms as well as the advantages and disadvantages they present. According to multiple sources [1] , two platforms stood out, **TensorFlow** from Google and **PyTorch** from Facebook which are the most used platforms and the best equipped for Deep Learning. In view of this comparative study between the two platforms, it emerged that despite the multiple advantages of PyTorch, TensorFlow would be more suitable for industrial use and especially for deployment towards micro-services. It has an integrated service that facilitates deployment on a micro-service.

The second exploratory research stage was to compare the existing pre-trained models architectures for Semantic Segmentation. There is a lot of architectures [2] [3] like

- **Fully Convolutional Networks FCN (2015)**
- **ParseNet (2015)**
- **FPN Feature Pyramid Network (2017)**
- **PSPNet (2017) : Pyramid Scene Parsing Network**
- **Mask R-CNN (2017)**
- **DeepLabv3+ (2018)**

According to different performances with differents datasets, the best architecture on indoor scene is the **Pyramid Scene Parsing Network (PSPNet)** [4].

This model, as shown in Figure 1, is composed of many blocks of neural networks layers, which given an input image (a), we first use CNN (Convolutional Neural Network) to obtain the feature map of the last convolutional layer (b). Then, a pyramid parsing module is applied to harvest different sub-region representations, followed by upsampling and concatenation layers to form the final feature representation, which carries both local and global context information in (c). Finally, the representation is fed into a convolution layer to get the final per-pixel prediction (d).

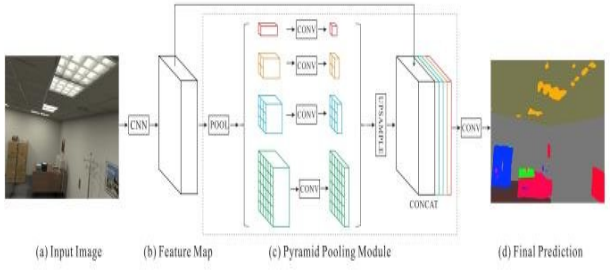

Figure 1. PSPNet architectures

2.2 Model Conversion

The second aim of this project was to convert a model from a framework to another. An image segmentation model obtained by subcontracting with another company The model provided was on the **Caffe** platform which is not as convenient for uses, is yet very efficient for integration into embedded systems such as mobile devices.
It was therefore necessary to convert this model from the Caffe platform to the TensorFlow platform which, at the end of our study of the different platforms was the best suited. A model being a set of layers of neurons and these are implemented in a different way depending on the platform used. Converting a model from one platform to another is therefore rewriting the layers so that they are understood by the target platform.
This part of the project required detailed research and deep understanding of the both frameworks and how to overcome the difficulties presented by the Caffe platform, particularly the correspondence between the different layers of neural networks.

2.3 Model Training & Deployment

Model training begins with collecting data from various open-sources [5] and pre-processing them in order to eliminate poorly annotated images or those that may have a negative impact on model performance.

The learning phase consists in passing the images several times in the model so that it updates by backpropagation the various weights of the model to recognize the 9 classes chosen for our model. However, training the images requires GPU resources that are optimized for operations on images that are treated as matrices.
It was then necessary to use a service specializing in cloud computing, **Amazon Web Services (AWS)** to obtain the necessary resources for training.
The table 1 presents the different times taken during training by our model and that taken by the initial model developed by the researchers and the 2 the differents training configurations used [6].

Table 1. Training configurations comparisons on PSPNet

Model	DS (k)	Time (h)
PSPNet (trained)	40	78
PSPNet (original)	20	14

Table 2. Training configurations comparisons

Model	Instances	GPU
PSPNet (trained)	1	16
PSPNet (original)	8	10

The deployment of the model in the cloud was abandoned due to the performance of the model deemed insufficient for an application to be put into production.

3 Results & Future applications

The results obtained after the conversion were poor. The converted model could barely recognize an image. After investigations these results can be explained by the lack of efficiency of the model conversion.
For the performance evaluation we used 3 metrics [3]:

- **Pixel Accuracy (PA)** simply finds the ratio of pixels properly classified, divided by the total number of pixels. For $K+1$ classes (K foreground classes and the background) Pixel Accuracy is defined as:

$$PA = \frac{\sum_{i=0}^{K} p_{ii}}{\sum_{i=0}^{K} \sum_{j=0}^{K} p_{ij}} \quad (1)$$

where $_{pij}$ is the number of pixels of class i predicted as belonging to class j.

- **Mean Intersection over Union (MIoU)** is the average of IoU over all the classes. IoU is also one of the most commonly used metrics in semantic segmentation. It is defined as the area of intersection between the predicted segmentation map and the ground truth, divided by the area of union between the predicted segmentation map and the ground truth:

$$IoU = \frac{A \cap B}{A \cup B} \tag{2}$$

where A and B denote the ground truth and the predicted segmentation maps, respectively. It ranges between 0 and 1.

- **F1 Score** is defined from Precision and Recall which are defined as follow:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

where TP refers to the true positive fraction, FP refers to the false positive fraction, and FN refers to the false negative fraction.

$$F1 = \frac{2PrecRec}{Prec + Rec} \tag{5}$$

The results of the training phase are shown in the following table. It is noted that the model trained to performance superior to the initial model. However, improvements would be possible when we notice that the model sometimes confuses certain elements such as the floor and the wall.

Table 3. Performance of the model

Model	mIoU (%)	Pixel Acc. (%)
PSPNet (trained)	47	83
PSPNet (original)	41	80

As the results are not perfect, we can consider improving the results by increasing the quantity of images and also exploring differentiation techniques between the classes combined.

One of the future applications of the model in addition to the deployment would be the use of the model for a real-time application allowing diminished reality for furniture.

4 Conclusion

This abstract presented the different stages of a machine learning project for image segmentation. The conversion step of the model did not bring satisfactory results. Due to this, it was necessary to carry out the training on the destination platform. The most important step was the selection of the architecture, datasets and metrics and the preprocessing of the images because they allow to considerably improve performance on the images.
Such a segmentation model could in the long term make it possible, thanks to a real-time application, to perform erasure to test the purchase of new furniture in a room.

5 Acknowledgments

We would like to thank all those who made this internship both an enriching and interesting experience. Firstly all the members of INSPI, Mr. Stéphane Mercier for the technical support and the opportunity to work on this project, the internship tutor Mr. Julien Fayer for all the advice and the supervision and Mrs Géraldine Morin for recommending me to Mr. Mercier.

References

[1] Kirill Dubokikov. Pytorch vs tensorflow, spotted the difference. https://towardsdatascience.com/pytorch-vs-tensorflow-spotting-the-difference-25c75777 2017.

[2] Arthur Ouakine. Review of deep learning algorithms for image semantic segmentation. https://medium.com/@arthur_ouaknine/review-of-deep-learning-algorithms-for-image-semantic- 2018.

[3] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. https://arxiv.org/pdf/2001.05566.pdf, 2020.

[4] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017.

[5] Hayko Riemenschneider. Yet another computer vision to index dataset. http://yacvid.hayko.at/, 2011-2020.

[6] Hengshuang Zhao. semseg. https://github.com/hszhao/semseg, 2019.