# Lab 13: “Fully Convolutional Networks for Image Segmentation”

Julio Nicolás Reyes Torres
jn.reyes10@uniandes.edu.co

Juan David Triana
jd.triana@uniandes.edu.co

*Computer Vision*
*Biomedical Engineering*
*Universidad de los Andes*
*2019*

## Description

In this lab are presented two architectures of *fully convolutional networks* for image segmentation over Pascal VOC database. In figure (1) is showed the implemented architecture and in figure (2) an example of ground-truth provided by Pascal VOC. The implementation is based on the article “Fully Convolutional Networks for Semantic Segmentation” by Long et al, which introduce a FCN end-to-end for pixelwise prediction and from supervised pre-training. [1]

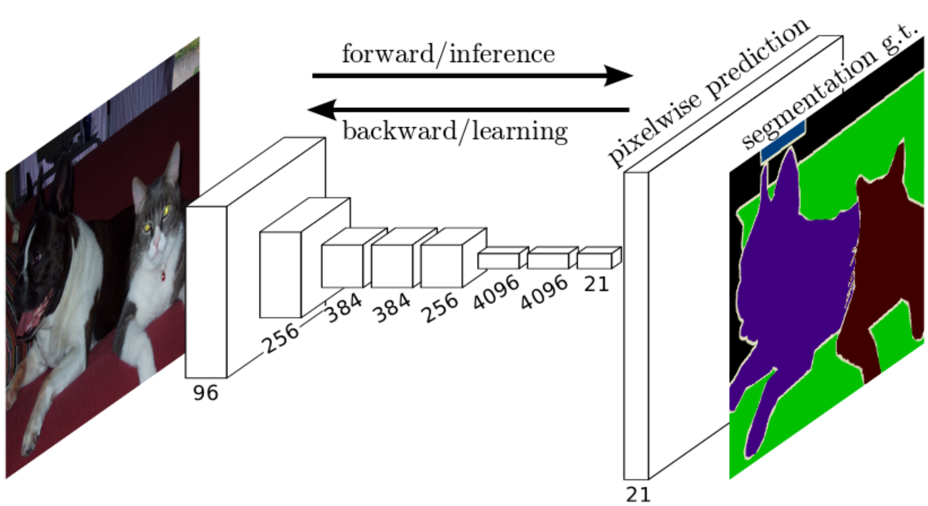



Figura 1: Architecture of FCN model.

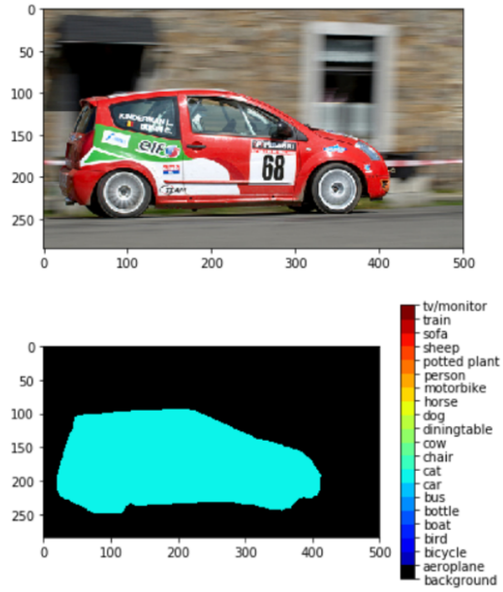



Figura 2: Example of image Ground-truth.

# 1. Architectures

The presented fully convolutional network has the property of learning the information in high layer with fine, lower layers. In figure (3) is presented the specific architecture with the combination of its layers. The prediction and pooling layers are presented as grids and the intermediate are shown as vertical lines. In the first row (*FCN-32s*) the single-stream net which upsamples stride 32 predictions back to pixels in a single step. In the Second row (*FCN-16s*) the net is predicting finer details because of combine predictions from the final layer and the pool4 layer, that is, retaining high-level semantic information.

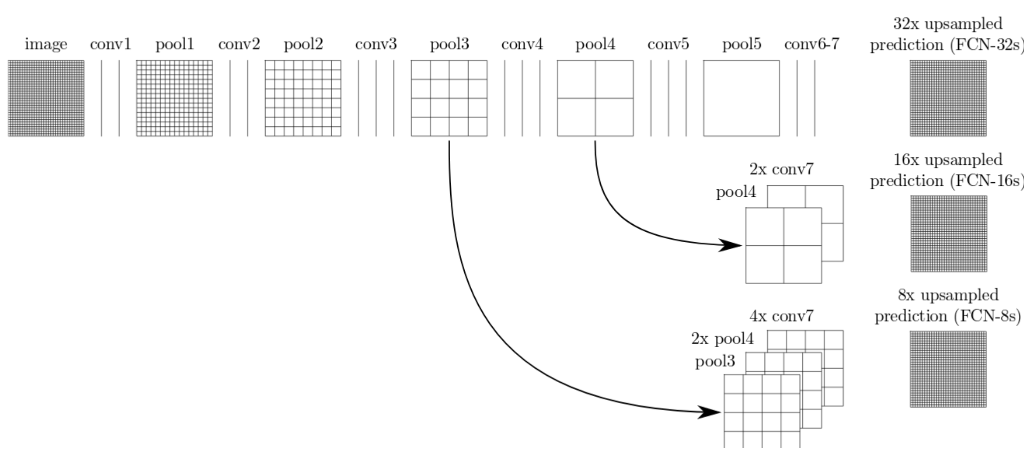



Figura 3: Model process to obtain 32s and 16s architectures.

## 2. Architecture FCN 32s

1. Layer 1
   - (64,3,500,375) 3x3Convolution
   - (64,64,500,375) 3x3Convolution
   - (64,64,500,375) 2x2MaxPool
2. Layer 2
   - (64,128,225,188) 3x3Convolution
   - (128,128,225,188) 3x3Convolution
   - (128,128,225,188) 2x2MaxPool
3. Layer 3
   - (128,254,112,94) 3x3Convolution
   - (254,254,112,94) 3x3Convolution
   - (254,254,112,94) 2x2MaxPool
4. Layer 4
   - (254,512,56,47) 3x3Convolution
   - (512,512,56,47) 3x3Convolution
   - (512,512,56,47) 2x2MaxPool
5. Layer 5
   - (512,512,28,24) 3x3Convolution
   - (512,512,28,24) 3x3Convolution
   - (512,512,28,24) 2x2MaxPool
6. FC1
   - (512,4096,16,12) 3x3Convolution
7. FC2
   - (4096,4096,16,12) 3x3Convolution
   - (4096,21,16,12) 1x1Convolution
   - 32 stride upsample

As it is seen on the architecture, the main focus of the segmentation task comes with the implementation of a 32 stride upsample. This means that all of the categories will be expanded from the final map to the initial size of the image, since it was downsampled 5 times or by $2^5 = 32$.

## 2.1. Result with VGG-16 pretained weights

First we implemented a training process with VGG-16 pretrained weights. After a series of iterations, the following image describes segmentation of the algorithm over a validation set.

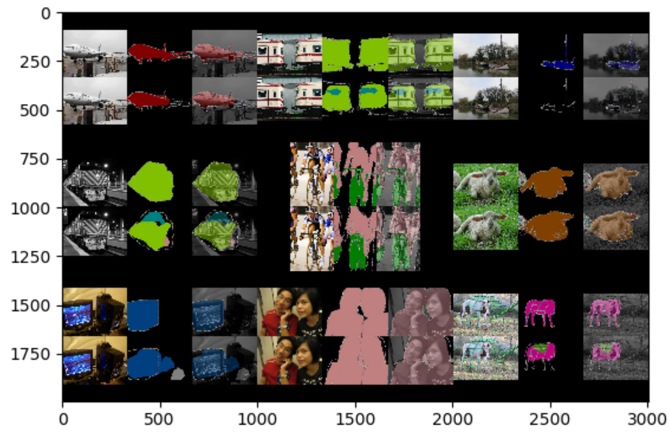
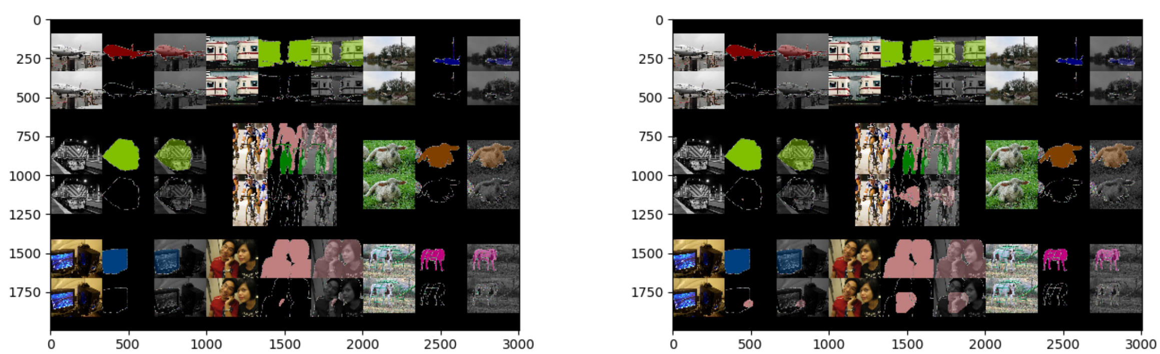



Figura 4: 32s with VGG pretrained weights (1 epoch).

## 2.2. Results from scratch (No fine-tuning)

Afterwards, a training process with no fine-tuning was implemented, as follows:

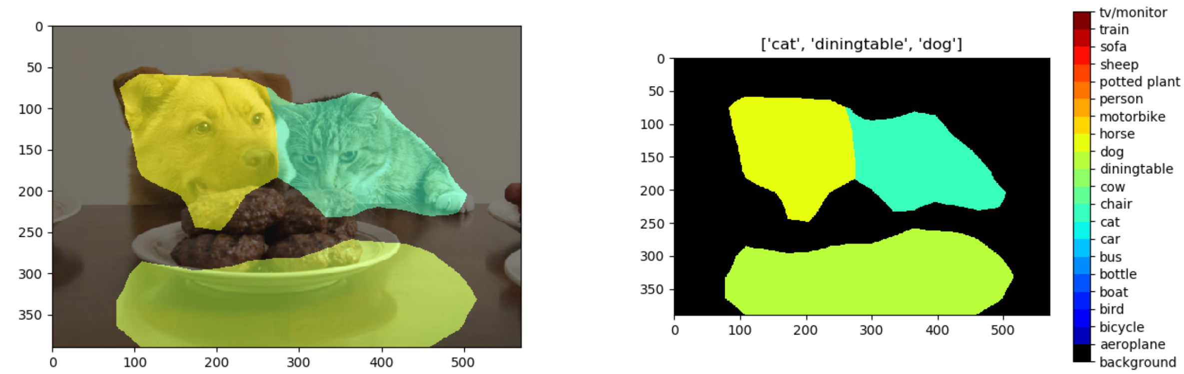



(a) Epoch 1

(b) Epoch 2

Figura 5: 32s Training from scratch.

Finally an implementation of the 32sFCN was made for images in the wild:




(a) Image segmentation

(b) Class segmentation

Figura 6: Image in the wild test of FCN 32s.

## 2.3. Discussion for 32s architecture

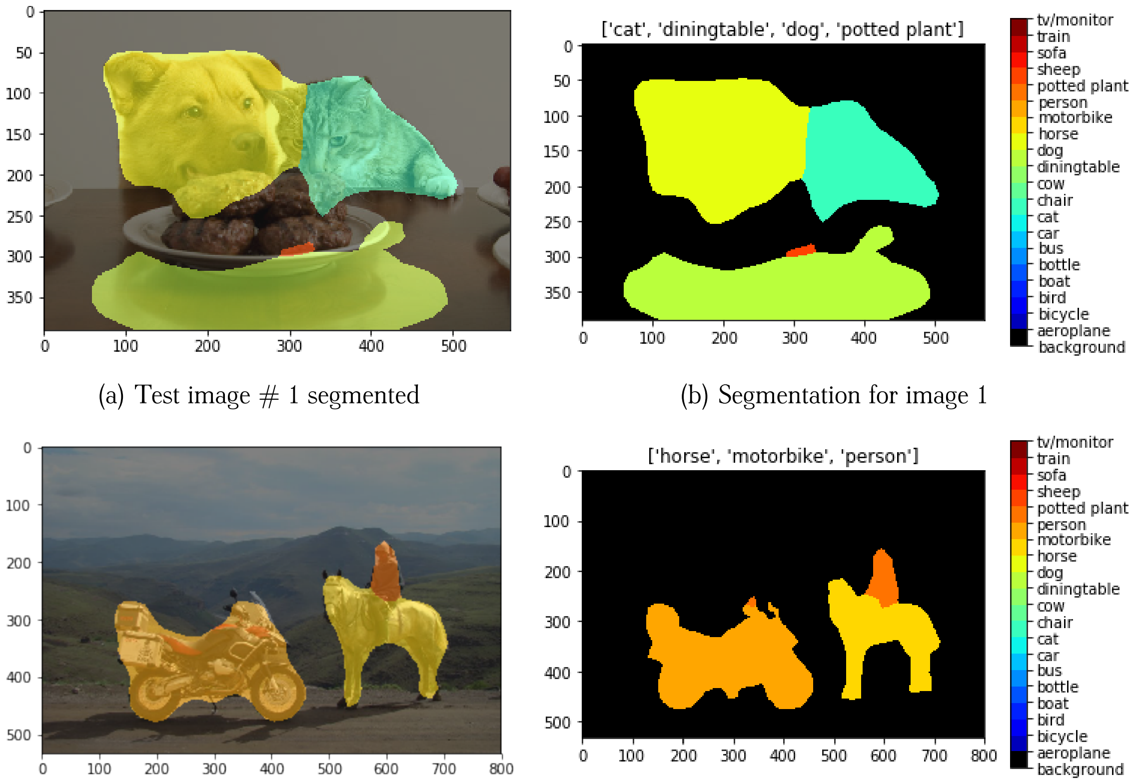
As it was seen on the results, fine-tuning is very useful and effective on segmentation tasks when it comes to small amounts of iterations. This is shown on figure 4 where an accurate segmentation is visualized on every single image on the first epoch of the algorithm. This is contrary to the results shown on figure 6, where a poor recognition of the object area was found on each image on epoch number 2.

# 3. Architecture 16s

The "fully convolutional net (FCN)" implemented for segmentation combines layers of the feature hierarchy and refines the spatial precision of the output [1]. The main difference on this architecture is to extract more information from "32x prediction" and use it to recover details that represent a better segmentation.

## 3.1. Results of segmentation with 32s weights

The results of train the net "16s architecture" are presented following:




(a) Test image # 1 segmented

(b) Segmentation for image 1

(c) Test image # 3 segmented

(d) Segmentation for image 3

Figura 7: **Segmentation with architecture 16s.** Image (a) Test image # 1, the image (b) shows the segmentation done for image (a). Image (c) is the test image # 3 and (d) its the corresponding segmentation.

## 3.2. Results from scratch (No fine-tuning)

The results of train the net 16s from scratch are presented following:

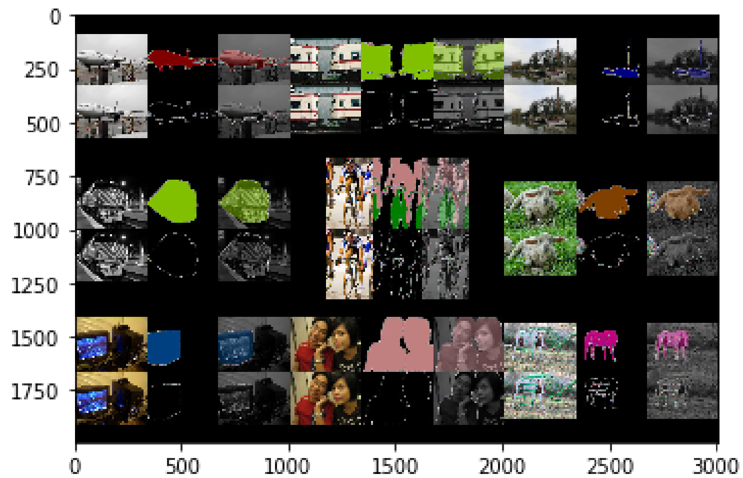
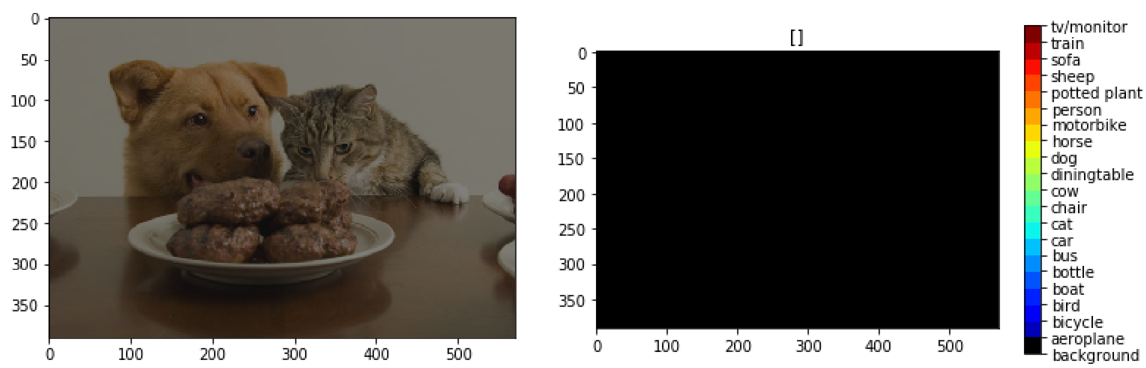



Figura 8: 16s Training from scratch.




(a) Image segmentation

(b) Class segmentation

Figura 9: Unsegmented image.

## 3.3. Discussion for 16s architecture

Figure (7) presents two final results of images segmented by FCN. Figure (a) shows a set of objects very clustered and partially represented, it is a difficult case, figure (b) is the segmentation. It is observed the objects are segmented and identified, however, the results are not the best. the contour is not totally segmented and an extra object (potted plant) is identify.

Otherwise, the test image in (c) and its segmentation (d) is the example of a good representation. The three objects are well identify (horse, motorbike, person) and very well segmented, the contours represents accuracy in the edges of the objects.

Finally, figures (8) and (9) present the result of training the model 16s from scratch. It can be observed that the model did not segmented anything, this can be understood because the net gets the information from 32s architecture, that is, if the net is trained from scratch it does not take anything and the model is not going to segment anything.

## 4. Conclusions

- The segmentation model presented with "architecture 16s" presents a better result than "32s model", in figures (7) and (6) are presented its segmentations respectively. This result is logical because the fully convolutional net (FCN) with architecture 16s for segmentation combines the previous layer of the feature hierarchy and refines the spatial precision of the output. Besides it is important to highlight, this model must take the previous weights from 32s layer, if this net is trained from scratch it is not going to segment anything.

- "Fully convolutional networks" represents a complete solution to obtain quality and accuracy in segmentation problems, specially the architecture with multi-resolution layer combinations, improves the learning and speed up the models.

## Referencias

[1] Long, J., Shelhamer, E., and Darrell, T., "Fully Convolutional Networks for Semantic Segmentation," Tech. Rep. [Online]. Available: https://people.eecs.berkeley.edu/{~}jonlong/long{_}shelhamer{_}fcn.pdf