POLITECNICO
MILANO 1863

ARTIFICIAL NEURAL NETWORKS AND DEEP LEARNING

(Prof. Matteo Matteucci, Prof. Giacomo Boracchi, Prof. Francesco Lattari)

TEAM The Neurons Burners

Second Assignment: Segmentation of Agricultural Images

Luca Colasanti

Stefano D'Angelo

Enrico Pancher

Academic Year 2020-2021

In this challenge we were asked to segment RGB images to distinguish between crops, weeds and background.

In this document we will try and explains the steps we underwent to reach the result.

Data Manipulation:

The first step was to address the subdivision of training data and masks in proper training and validation subsets, for both the actual images and their related masks. Refer to the notebook "Data_Manipulation_Colab" for further explanations.

Data Augmentation:

In order to maximize the effectiveness of the algorithms employed, we performed data augmentation on the training images and masks datasets, with obvious care that the same image and mask were applied the same transformations to. The transformations we applied were:

- Width shifts;
- Height shifts;
- Horizontal flips;
- Vertical flips.

All transformations that could cause "class mismatch" in the original dataset have been avoided (i.e. zoom and rotations, that we tried including in initial iterations, but they proved rather ineffective).

Model building:

We focused our efforts initially on the mais datasets (BipBip) and later moved on the haricot datasets, in particular the BipBip and Weedelec ones. The first approach in addressing the problem stemmed from the employment of the notebook provided by prof. Lattari's exercise sessions, in order to familiarize and get acquainted with the problem: we used a VGG16 CNN for the encoder part of the model and an UpSampling2D layer and a convolutional 2D layer with a ReLU activation function for the decoder part. Finally, we used a Conv2D layer for the prediction. Refer to the notebook "Multiclass_Segmentation_Tiling" for further explanations.

While analysing the first iterations of a solution arisen in this initial phase, we identified a problem related to a low performance of the model: performing resizing on the images reduces the details and greatly hampers the resulting masks. Consequently, we have decided to employ two new solutions for the model:

- Performing tiling on the images: rather than resizing images, we decided to perform tiling, that is we subdivide the images in smaller portions of a given size called "patches" (either 256x256 or 512x512 pixels patches), give them as input to the network one by one and then recompose the final mask stitching together the single masks identified using each patch. We mainly referenced the article "Tiling and stitching segmentation output for remote sensing: basic challenges and recommendations" and adopted an approach without clipping or overlapping of patches, since the datasets we mainly worked on allowed us to avoid having to perform averaging.
- Adopting the U-Net as the encoder: rather than the VGG16, we decided to adopt the U-Net based on the analysis of related papers. Its ease of use and applicability to many ranges of problems convinced us to use it in our solution.

The introduction of these changes is found in the notebook "UNet_Multiclass_Segmentation_Tiling". The main issue with this notebook was that it gave us back low results, thus we decided to rely on an alternative implementation of the UNet as presented in this article: https://towardsdatascience.com/unet-line-by-line-explanation-9b191c76baf5.

We also decided to introduce another term of regularization in the loss function, the momentum, in the case of tiling 1x1 with a batch size equal to 1.

Using this new implementation, we managed to elaborate the final notebooks used in the submission, divided by the dataset they work on specifically and obtain the results that are listed in the table below.

Model Training:

We trained various models, starting from all the models mentioned in the previous paragraph, but we focused on the last implementation on which we tried applying different types of tiling (1x1, 2x2, 4x4), with variable degrees of success, but without any significant prevalence of one alternative over the other ones. One noteworthy aspect has been that the smaller the size of the tiling, a little bit better were the performances of the network in terms of mean IoU.

Model Name	Dataset	TILING	PADDING	img_w	img_h	Data augmentation	Batch size	learning rate	Early stopping #epoch	Best Mean IoU (validation)	Best IoU #epoch	submission file	Accuracy Codalab
UNet_Dec18_20-48-32	BIP-BIP/MAIS		1	512	512	1	8	1,00E-04	7	0,6382	17	prediction_6	?
UNet_Dec19_01-36-43	BIP-BIP/HARICOT		1	512	512	1	8	1,00E-04	25	0,5305	27	prediction_7	?
UNet_Dec19_18-39-14	BIP-BIP/HARICOT		1	1024	768	1	4	1,00E-04					?
UNet_Dec20_01-51-28	WEEDELEC/HARICOT		1	576	576	1	8	1,00E-04	20	0,5613	19	prediction_8	?
	BIP-BIP/HARICOT	3X4	1			1		1,00E-04					?
UNet_Dec20_21-46-21	BIP-BIP/HARICOT	2x2	1			1	4	1,00E-04	32	?	(provato 32)	prediction_9	?
UNet(1X1)_Dec22_20-37-59	BIP-BIP/HARICOT	1X1	1			1	1(max senz	1,00E-04	31	0,6347	30	prediction_10	0,65
UNet(1X1)(NoPadding)_Dec24_13-37-06	BIP-BIP/HARICOT	2x2	0			1	4	1,00E-04	36	0,4958	36	prediction_11	
UNet(1X1)(NoPadding+Momentum0.20)_Dec25_00-20-20	BIP-BIP/HARICOT	1X1	0			1	1	1,00E-04	NO E.S.	0,0357	NON SONO STATE GENERATE IMMAGINI NE PREDICTION		
	BIP-BIP/HARICOT	2x2	1			1	4	1,00E-06					

Table 1. All trained models, with eventual submission results.

Final considerations and future developments:

As of the time of submission, we have not managed to resolve some issues related to the usage of padding: we have used a non-overlapping approach on the patches with a padding of zeroes. The outline of a patch was affected by this kind of approach and worsened the identification of classes, thus further developments should be focused on addressing this issue. Another possible optimization could be performing overlapping of tiles and averaging the results along overlapped portions of the patches to work around the issue of mismatch along the edges because of padding.