NAME\_1: Marco

LAST-NAME\_1: Introvigne

STUDENT-ID\_1: 10750466

NAME\_2: Antonio

LAST-NAME\_2: Urbano

STUDENT-ID\_2: 10527285

NAME\_3: Enrico

LAST-NAME\_3: Voltan

STUDENT-ID\_3: 10525467

LEADERBOARD NICKNAME: 3LN (Three Little Neurons)

Since the dataset consists of multiple companies (Weedelec, Bipbip, Roseau, Pead) and types (Mais and Haricot) we approach this task by going step-by-step. Below the main steps characterizing our design choices:

1. Understand the problem and the dataset

Before building the model, we split, as usual, our training dataset into training and validation sets, in order to better understand the performances of the models.

On the considered dataset we used the Data Augmentation preprocessing technique by applying some transformations on the images (like rotations, shear, zoom,...). For this purpose we started with no Data augmentation and then when we needed to reduce overfitting, we gradually increased the type and the value of the parameters in the ImageDataGenerator.

An important remark is about the dataset we used. At the beginning we started training the model on the entire datasets, i.e. all the companies and types. This gave use very poor results. For this reason we decide to split the training dataset either according to the types, i.e. training all the Mais and all the Haricot, or according companies. These two approaches gave us a better result.

So for this reason the last step was training eight different models over the eight possible combinations of companies and types. This approach gave us the best result, also because since each company has a different resolution, we could build the model according to the considered dataset.

Another important step to be remarked is about the resolution of the images. Indeed the images have an image size that is too big to be trained as they are. For this reason at the beginning we decided to apply the *resize* function and reduce the dimension.

Later we go through cropping the images in several blocks, training over the cropped parts and finally rejoin everything. This approach gave us better results than the previous.  
A final remark is about the crop method. Indeed at the beginning we split all the images in 4 equal square blocks, but when we noticed that the different companies may have different image dimensions we implemented a parametric crop in which we can decide the number of rows and columns and so we further improve our results

2. Models implementation

We have implemented two different Neural Network Architectures.

All the models, except for the Roseau company, have been trained using a Unet architecture which is described in the *Unet.py* file. After having built the network, we have performed some tuning on the parameters value (i.e. base\_filter, depth of the network and input image size) in order to reach the best result. For the architecture we have tried either with and without a BatchNormalization layer and for the decoder blocks either with an Upsampling layer or a Conv2DTranspose layer.

The best combination was the one using the BatchNormalization and Conv2DTranspose.

This model had a bad performance on the images of the Roseau company. For this company we adopted another approach, i.e. the first part of the network (encoder) has been initialized with VGG19 weights, then the remaining weights have been trained. After that, for the last epochs, we updated all the weights, also the pre-trained one, in order to further improve performances.

Due to the type of images, another important difference w.r.t the other companies was to apply a pre-processing function (from VGG19) on the input.

3. Submission

In the ‘*Official\_Submission*’ notebook we have created a dictionary in which we join the different results obtained by the single models. In particular for each company and type we defined the input image size, the number of rows and columns for the crop method and the actual model used for that dataset.

So for each combination of type and company, we define the ‘template’ of the model used during training, we load the saved weights and finally we perform the prediction.

A final step that allows us to slightly improve the prediction results is to rotate 90° each crop four times and compute the prediction as sum of the prediction of the four generated images.

4. Future work

The Roseau company was the most difficult one to predict, due to the type of images. As described in the previous steps to improve the performances for this company we have used a preprocessing function for the input and another model (w.r.t. the one used for the other companies) for training.

For Roseau we performed 2 further improvements that due to the the deadline we are not able to apply to the other companies:

* Shuffle the dataset after the end of each epoch
* Train only the cropped images which corresponding mask composed of all the three classes

**We didn’t put the weights of each model in the zip file for space reasons. If you need, ask us :)**