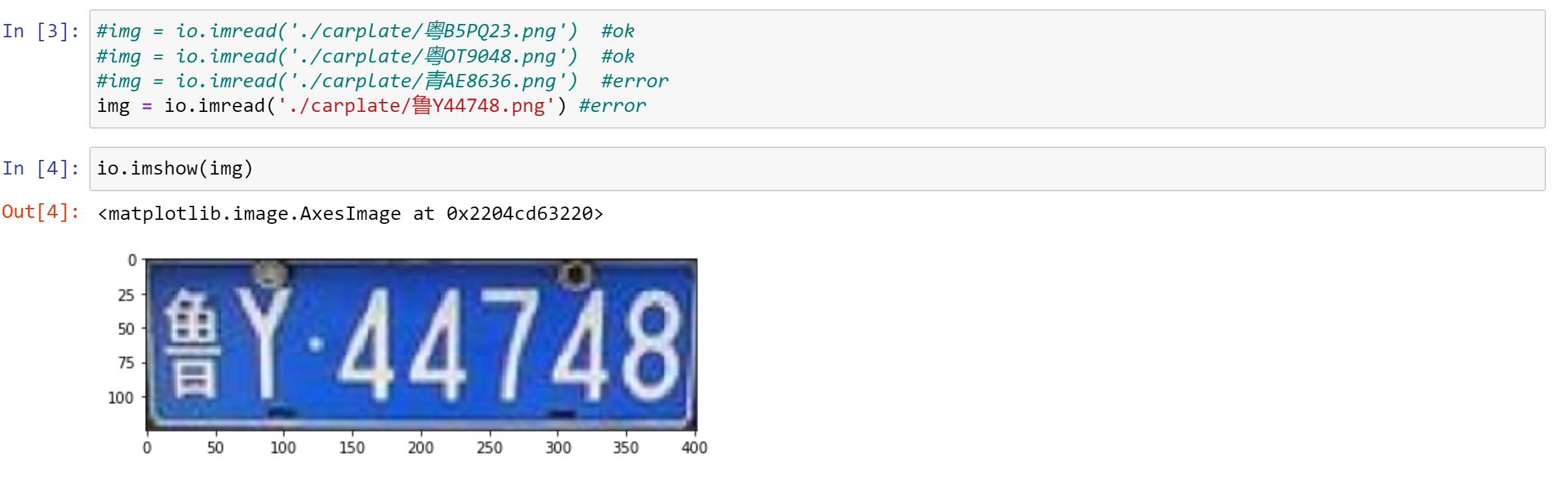
Report 5 - Chinese plate recognition problem

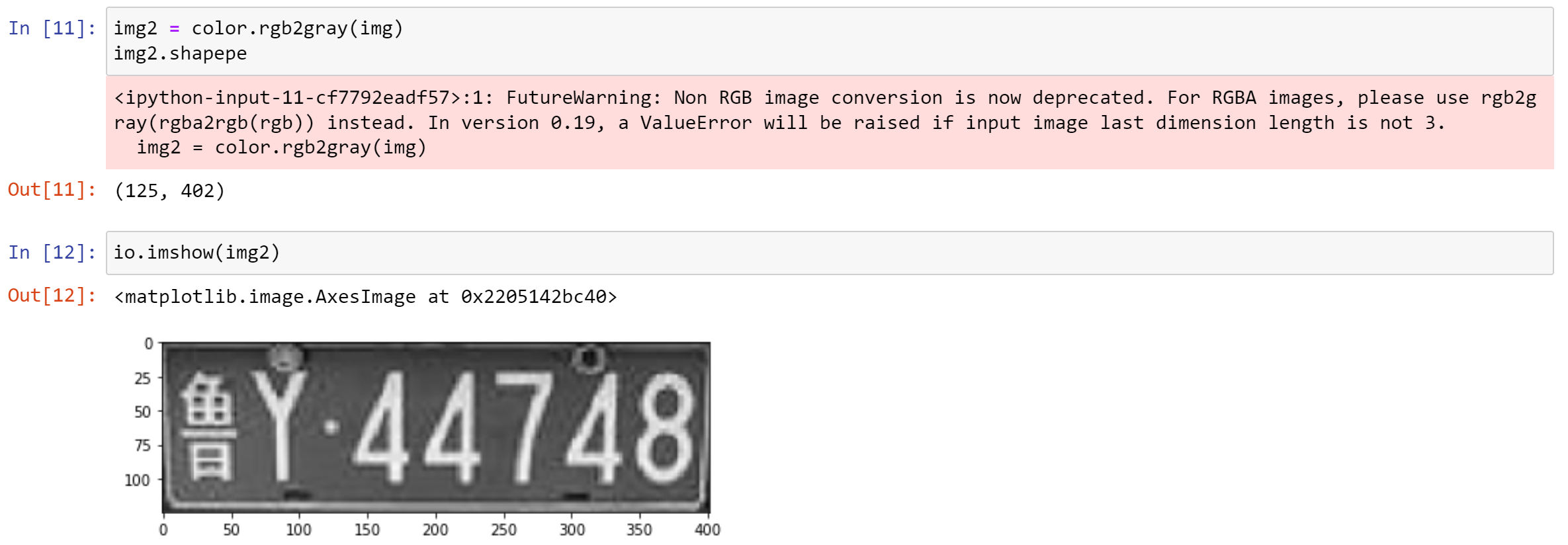
Jack 1930026143

**Basic working flow**

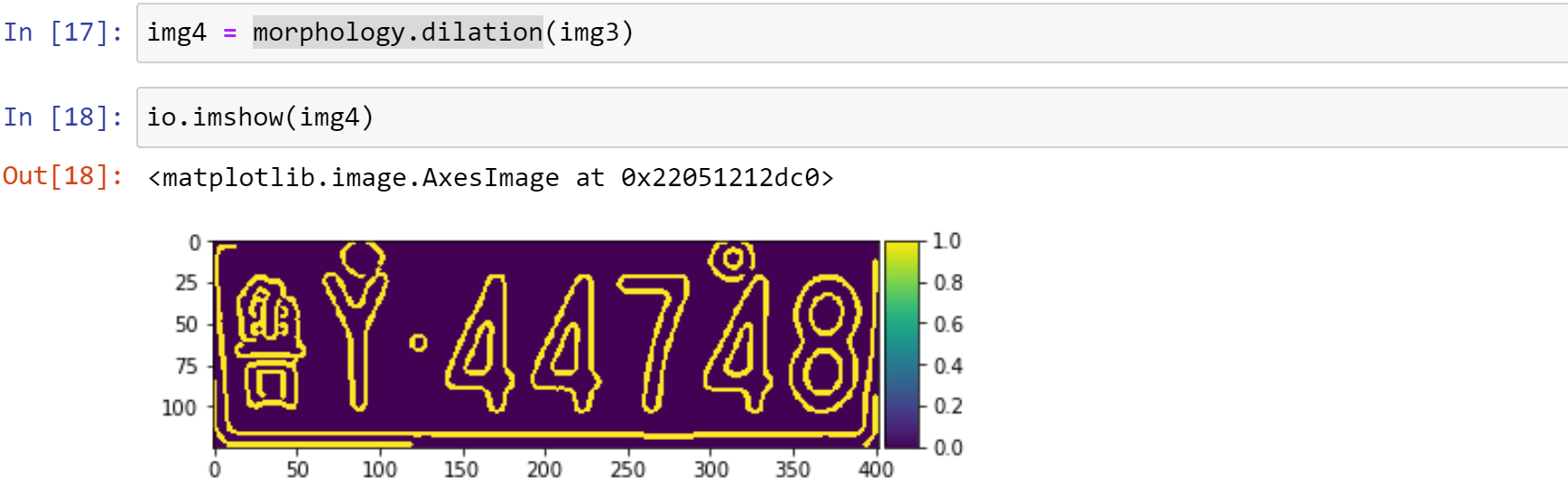
1. Load the image data



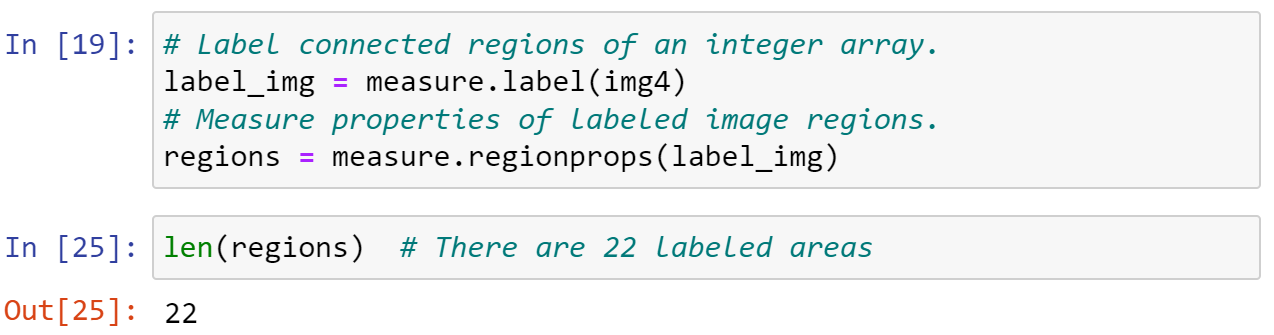
1. Transform to Grey Image, we can use the machine learning package and transform directly. And we can find that the shape of the image is from 3 dimension (length \* width \* RGB channel) to 2 dimension.



1. Apply Canny Edge Detection and Dilation. Firstly, we use the method to find the edge and use an useful package method to make the edge more obvious.



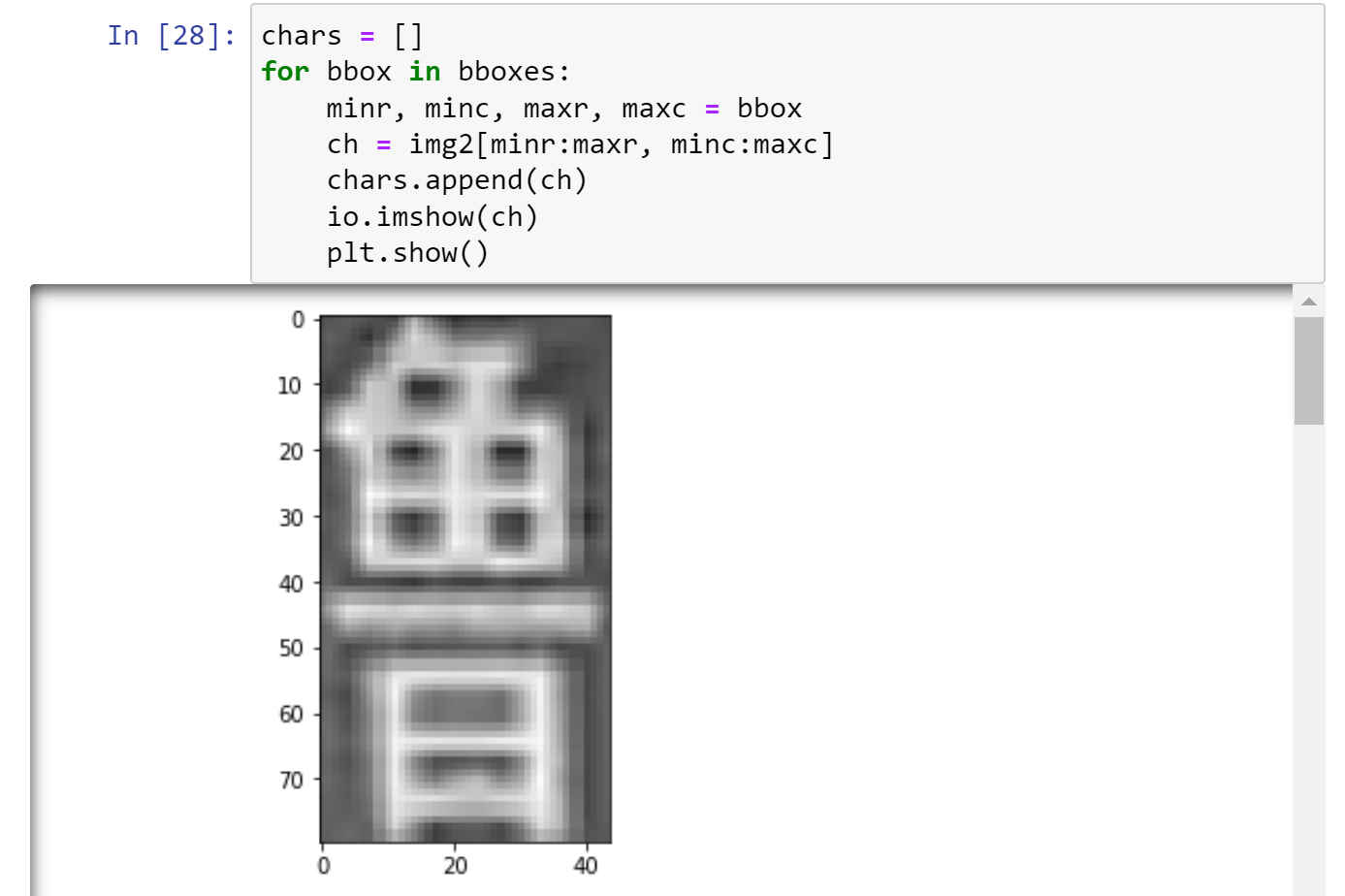
1. Label and Region Proposal. Firstly find the labeled connected regions of an array by the method of . Then in order to get the information and the properties of this region, we should use the . Print the length of the regions we know that the image has 22 labeled regions.



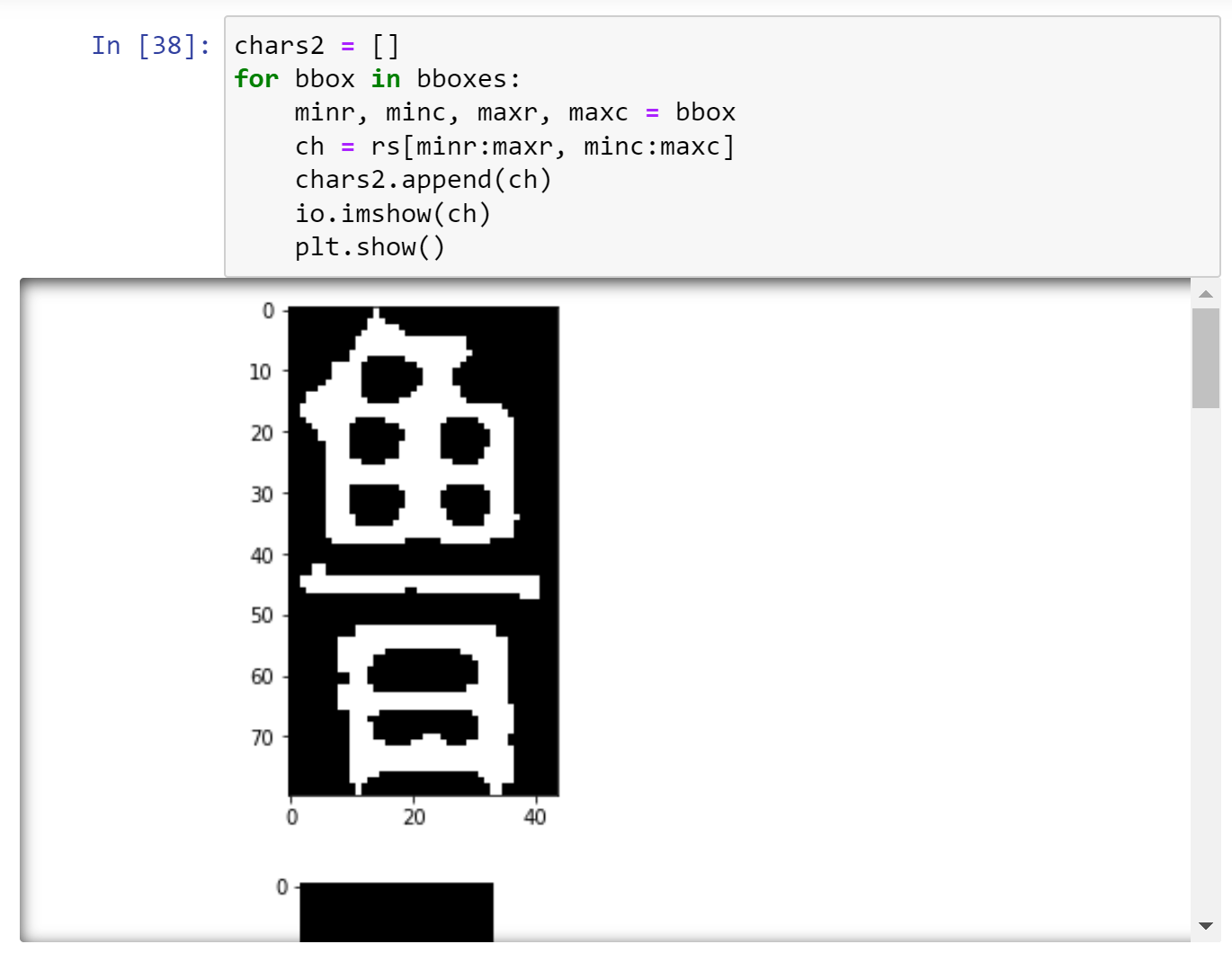
Then, we should filtrate these regions. For each labeled image area, if it is out of the limitation the scope of the segmented region, or it go beyond the border, we should remover the region. In addition, There are two circular identification knobs on the license plate which may be recognized into the region, we should also remove it. After that, the left regions are the features we want to obtain.



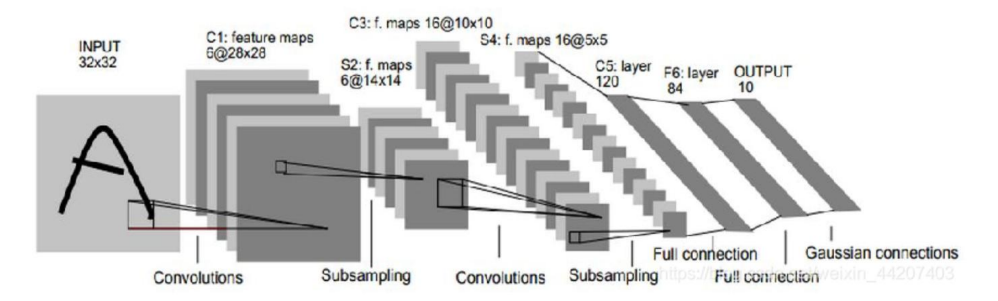
1. Retrieve Each Character Image. By the coordinate of the four corners of filtered area, we can get each image.



1. Image Enhancement. In order to improve the accuracy and convergence speed of the model, we can input the image data of the model for enhancement, which means that make the features of the image more prominent. We use the cv2 package to normalize and scalarize the data to implement.

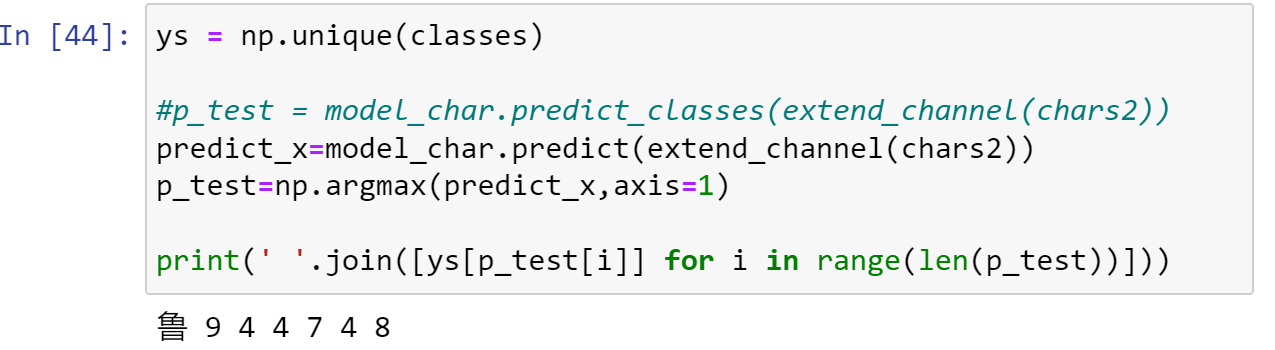


1. Character Recognition. Firstly, we should build the model. We choose the Convolution Neuro Network () model. it is very suitable to deal with the images data and it is very similar to the model.



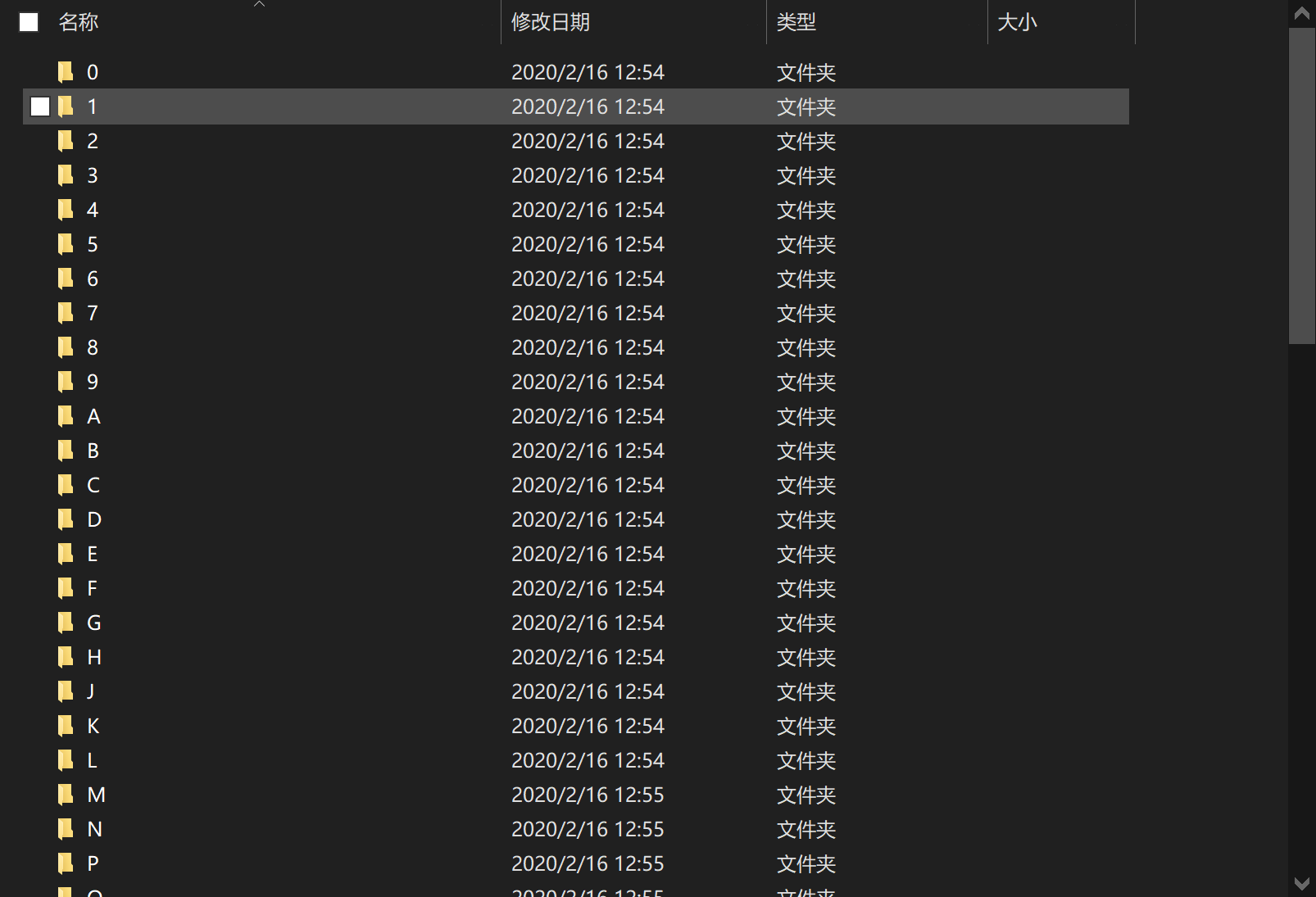
We can find that it has two convolution-pooling layers and some full connection layers. After training the model we can get the possibility of each label, which means that we can finish the images classification. After deploying the model by the method in , we just compile it and get the initial parameters. Here we do not train by this model again and just load the parameter weights by from the local disk.

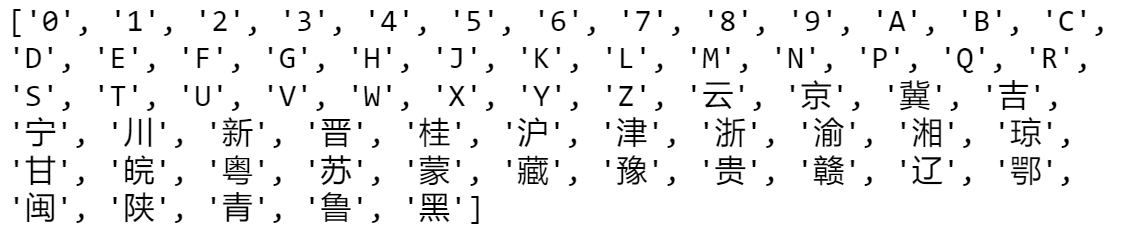
After that, we can task the model and print out the predicted result.



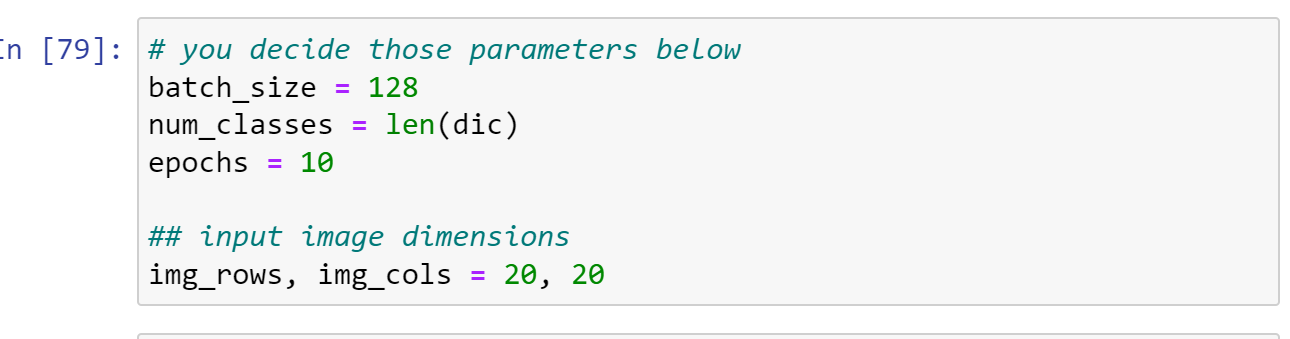
**Get the result by train the model.**

1. Load character data from the dataset subfolder. Under the dataset path, there are many subfolders which includes the images dataset of the folds name. The image is the in the model and the name of the folder is label of the image.

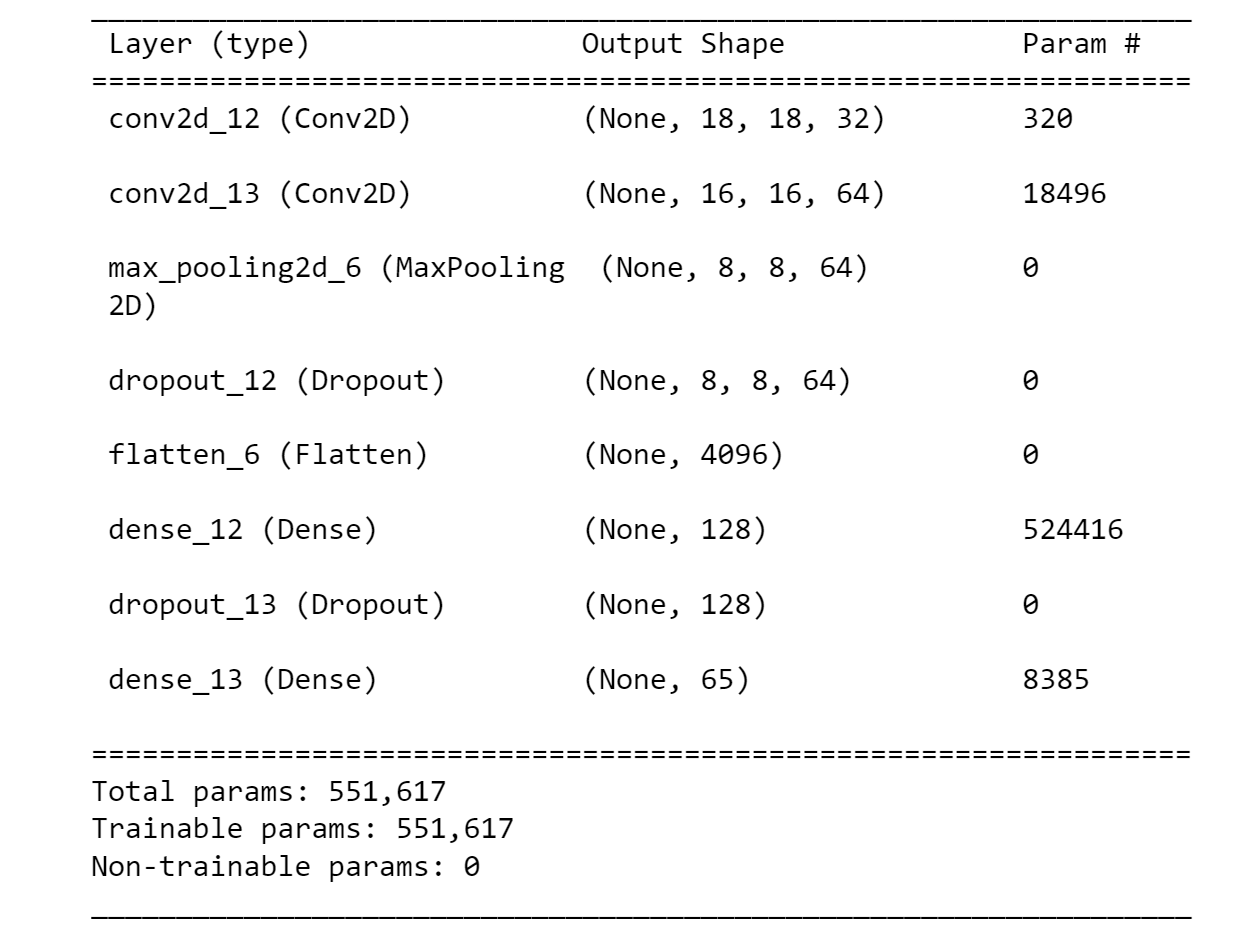




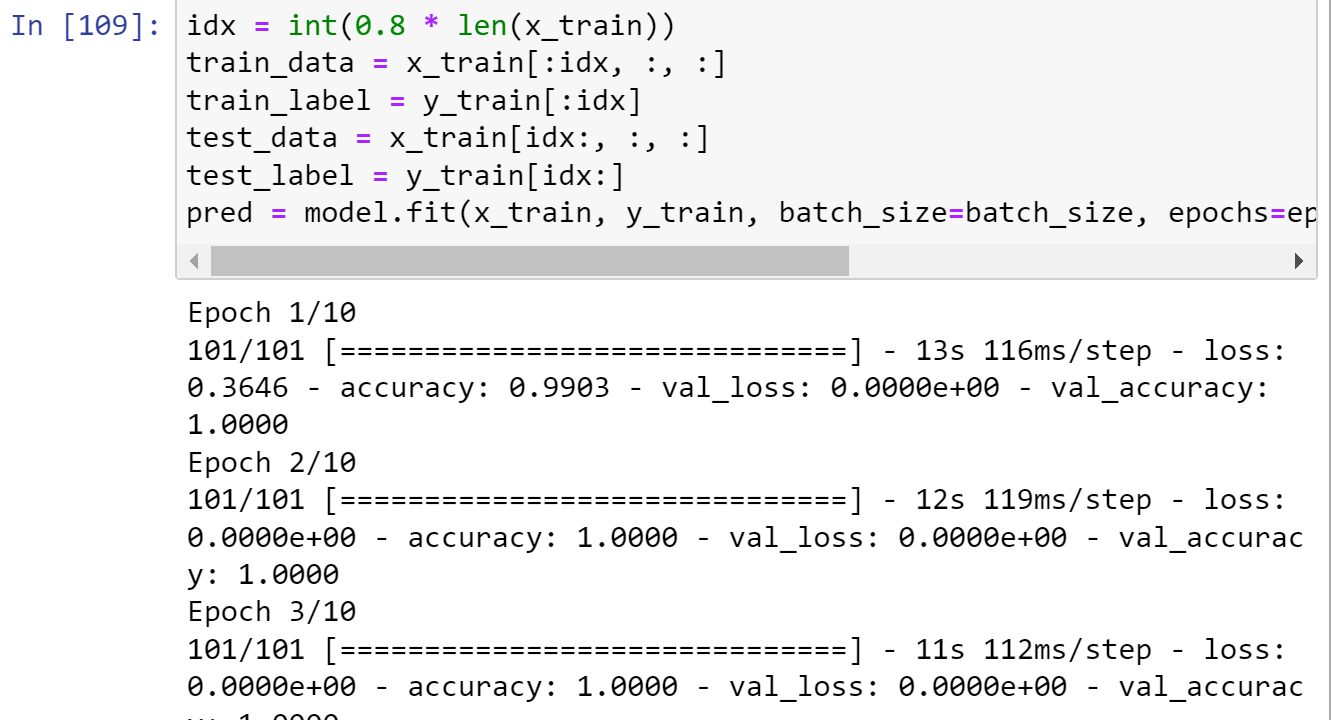
1. After processing the image just same as the above (GRB to gray, Edge Detection and Dilation) , we should divide the data to the training set and the validation set.
2. Decide a few parameters for training and extend the data channels to 4 dimensions



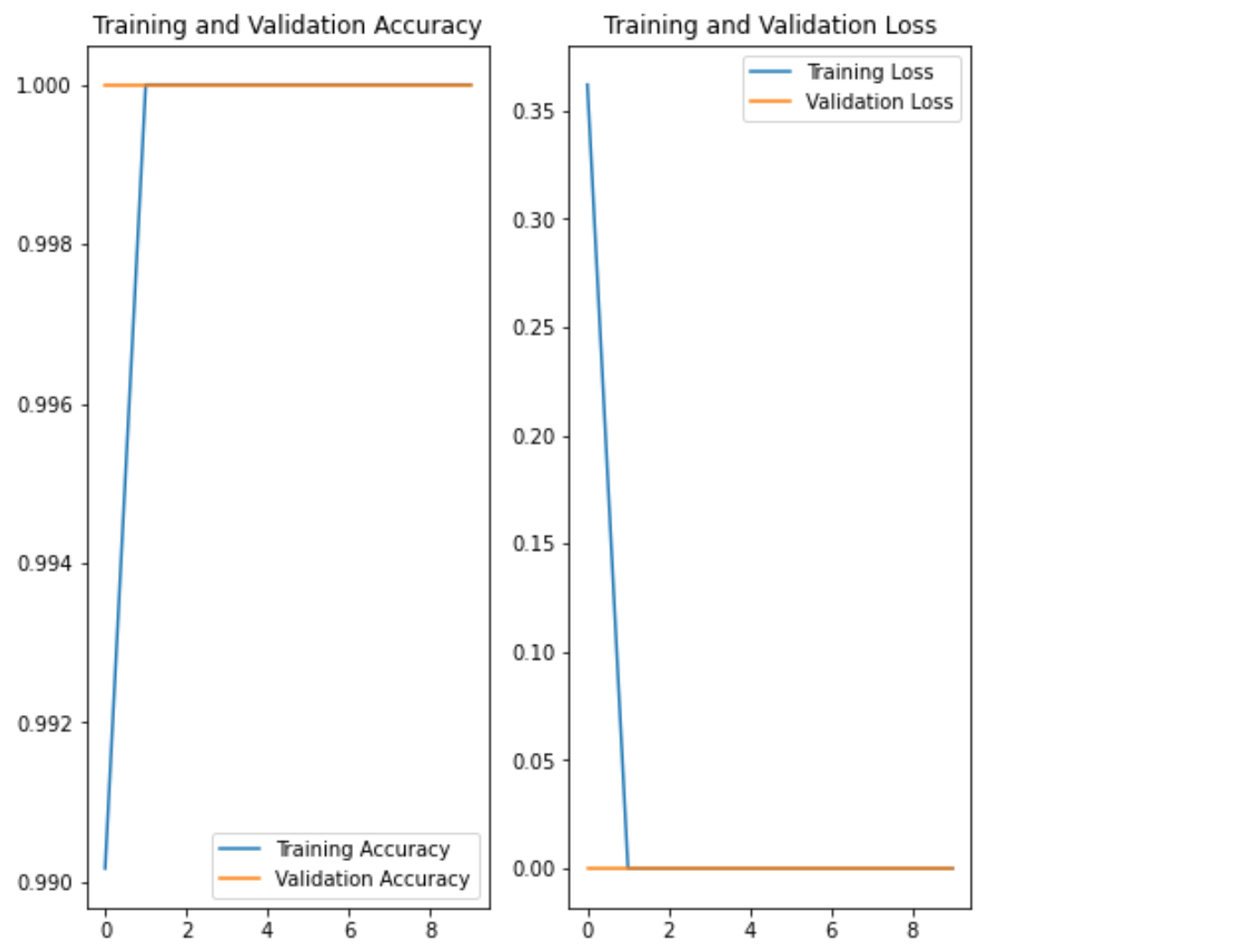
1. Before training the model, we should do the data normalization to help our model to convergence. The model is the CNN model which is introduced before. Compile the model and print the detail by the method.



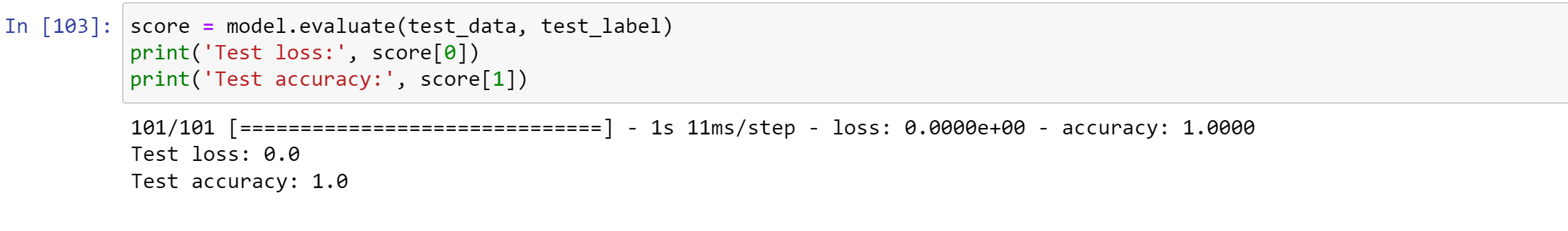
1. Train the model by using the test data and training data. We can find that at the second stage, the loss has gone down to zero



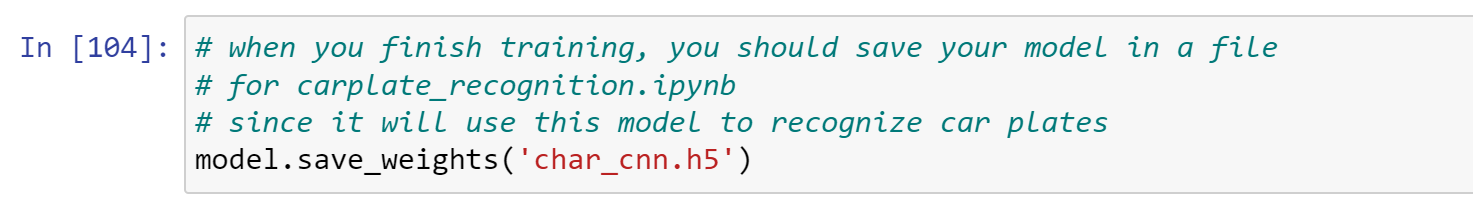
1. Visualize training results, we can see the accuracy and loss of the process of training:



1. Evaluate the model with test dataset. The model is good enough to be 100 percent accurate for the test set. So we don't have to fine-tune the model or deal with the overfitting.



Finally, just save the weight of the trained model so that next time to use.



**Difficulties Encountered**

The biggest problem is to fit the model. By plotting the training history, I found that both loss and accuracy on validation set and training set are presented as fluctuations are high (From 92% to 98%). I modified some hyperparameters, however, after many tweaks, the result is not changed obviously. Then I do the dilation operation and encounter the training set. In addition, I check the label definition and find that the test set form. Then after dealing with that, after only two iterations, the accuracy can reach 99% at least.