Instituto Tecnológico de Aeronáutica – ITA
Inteligência Artificial para Robótica Móvel – CT-213
Professor: Marcos Ricardo Omena de Albuquerque Maximo
Aluno: Lucas da Silva Jorge

Lecture 9 Lab – Convolutional Neural Networks

Methodology

The Convolutional Neural Network implemented in this activity, Lenet-5, was developed to recognize handwritten and machine printed characters. The adopted version of Lenet-5 is described in the following table:

# Camada	Tipo	Número de Filtros	Tamanho da Saída	Tamanho do Kernel	Stride	Activation Function
Entrada	Imagem	1	32x32	-	-	-
1	Conv2D	6	28x28	5x5	1	tanh
2	AveragePooling2D	6	14x14	2x2	2	-
3	Conv2D	16	10x10	5x5	1	tanh
4	AveragePooling2D	16	5x5	2x2	2	-
5	Conv2D	120	1x1	5x5	1	tanh
6	Dense (FC)	-	84	-	-	tanh
7	Dense (FC)	-	10	-	-	softmax

Table 1: architecture of Lenet-5 network.

The input layers has dimensions (32, 32, 1), which stands for 32 bits of width, 32 bits of height and 1 dimension of color, as exemplified in the figure below:

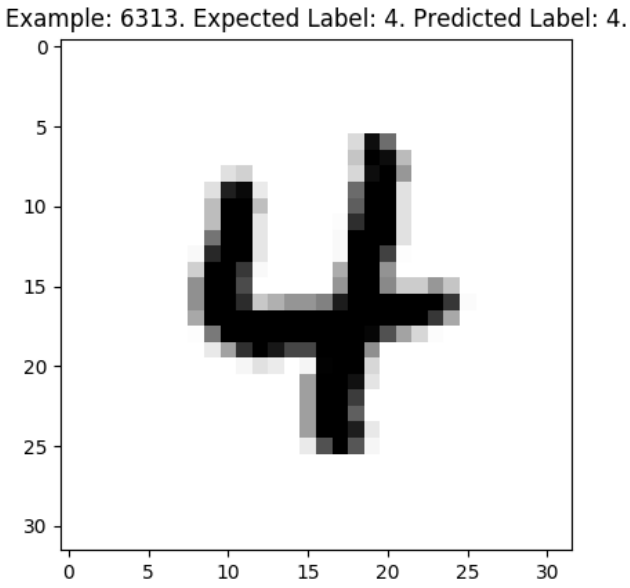
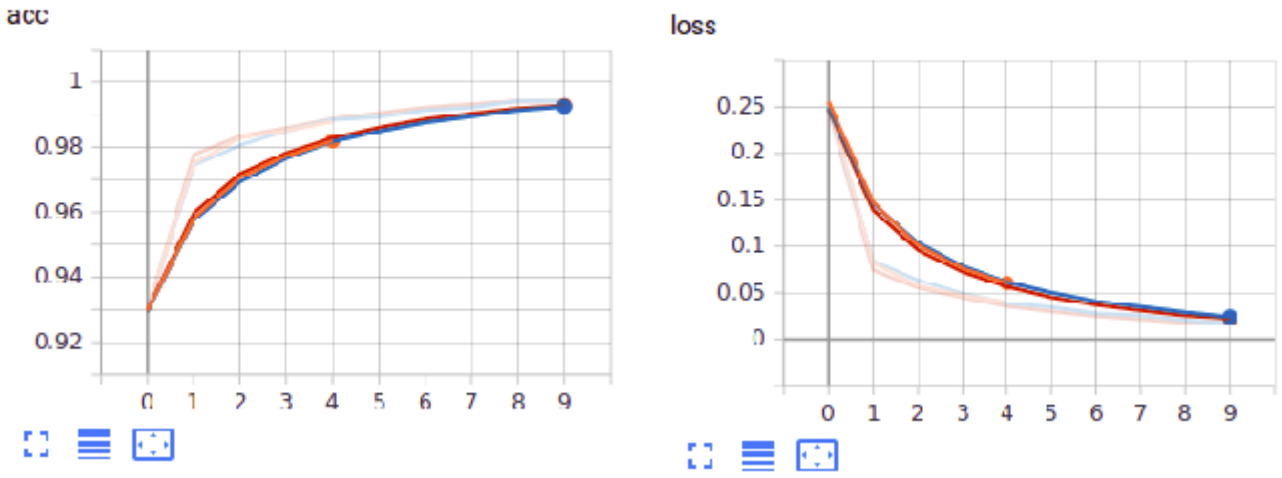


Figure 2: image with dimensions (32, 32, 1)

To train the network, we use a training dataset called MNIST, which contains 60000 training images for handwritten characters, and 10000 test images. Each training sample is a handwritten character and the corresponding accurate expected character.

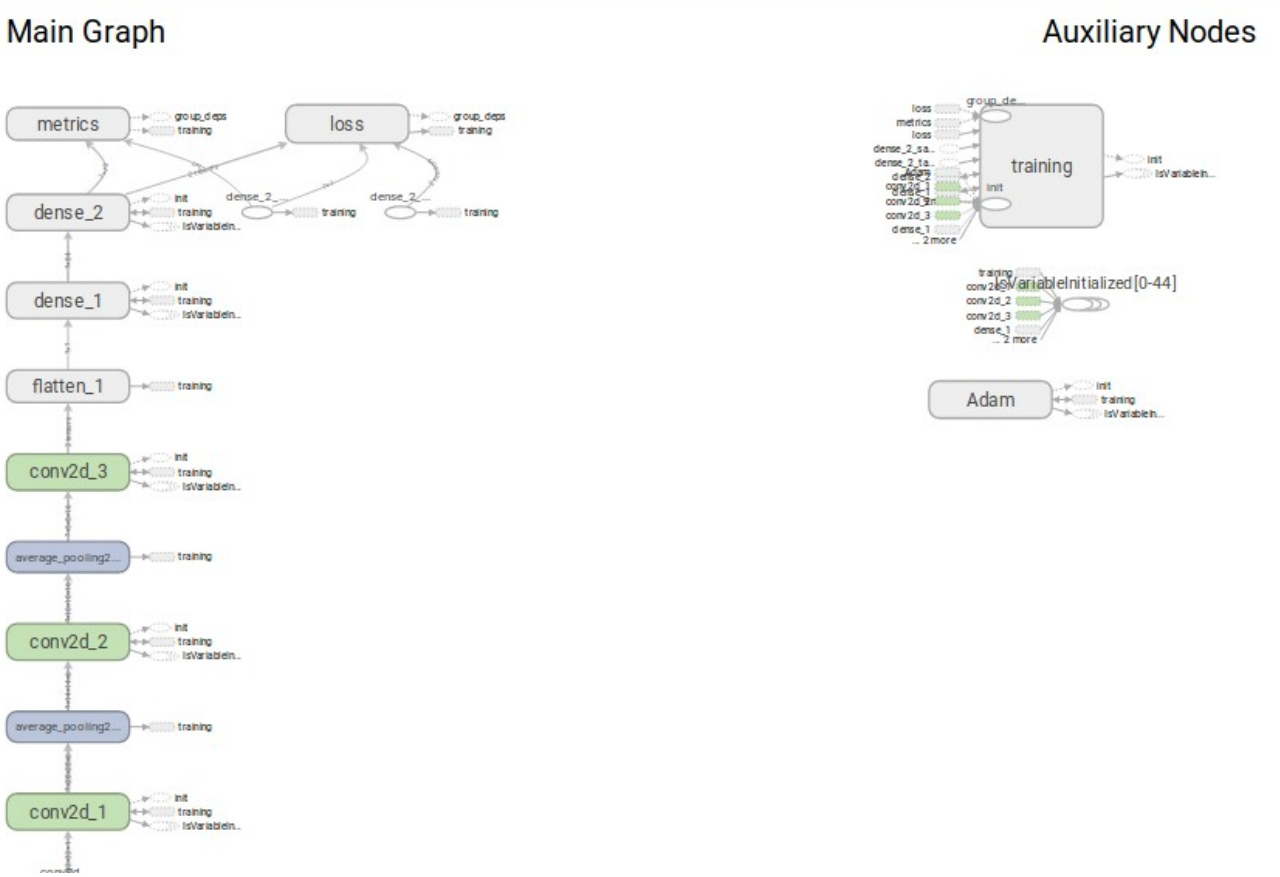
Results

The evolution of the accuracy and loss in the network along the training process are represented in the figures below:

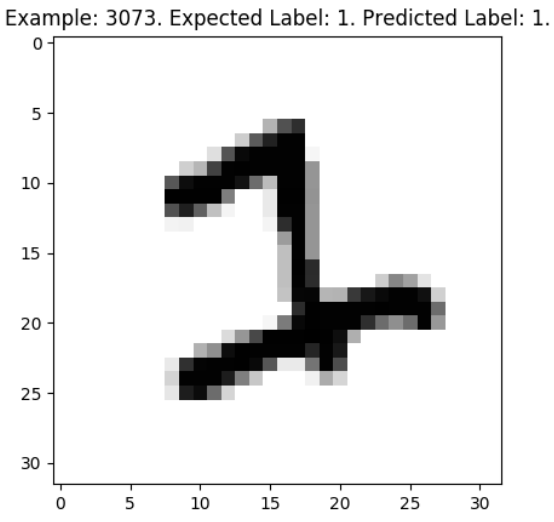


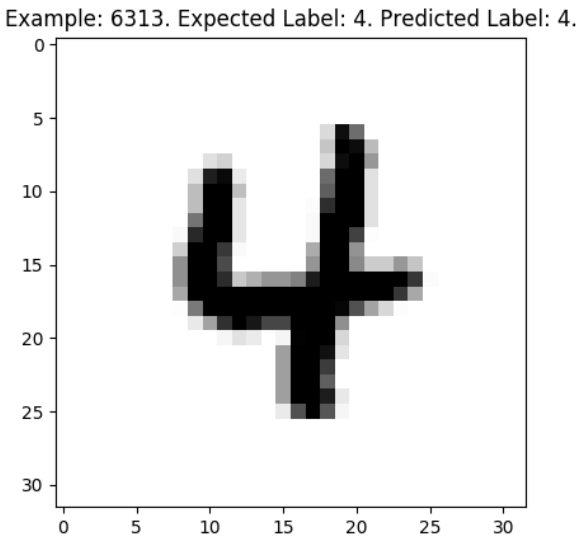
The accuracy grows and the loss decreases.

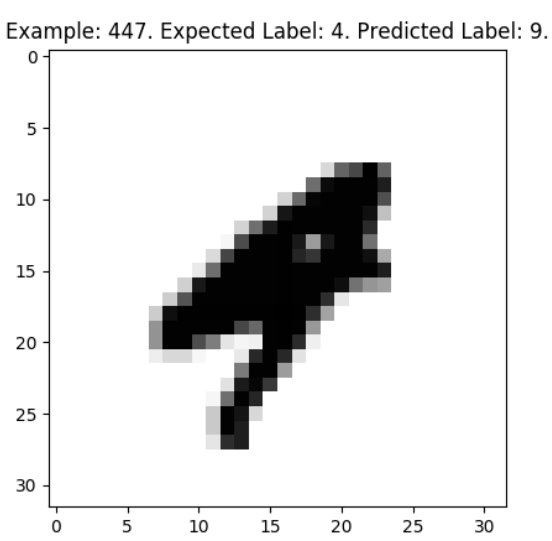
The resulting graph is the following:

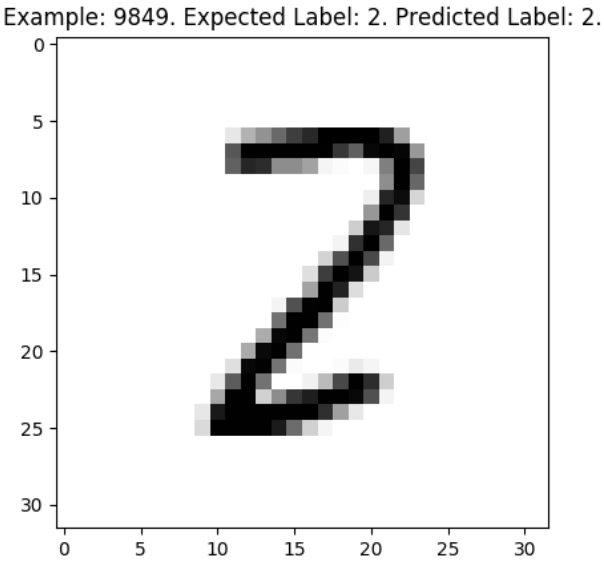


Finally, after trained, the CNN was used to evaluate some handwritten characters. Some predicted/expected results are listed below:









Actually, the case presented in which the prediction failed (expected=4, predicted=9) would easily be confused by humans as well.

The final accuracy obtained was 98.89%:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 28, 28, 6)         156
_________________________________________________________________
average_pooling2d_1 (Average (None, 14, 14, 6)         0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 10, 10, 16)        2416
_________________________________________________________________
average_pooling2d_2 (Average (None, 5, 5, 16)          0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 1, 1, 120)         48120
_________________________________________________________________
flatten_1 (Flatten)          (None, 120)               0
_________________________________________________________________
dense_1 (Dense)              (None, 84)                10164
_________________________________________________________________
dense_2 (Dense)              (None, 10)                850
=================================================================
Total params: 61,706
Trainable params: 61,706
Non-trainable params: 0
_________________________________________________________________
10000/10000 [==============================] - 6s 629us/step
Test loss: 0.03243423487313558
Test accuracy: 0.9889
lucas@lucas-HP-14-Notebook-PC:~/Desktop/ITA/CT_213/lab_9/lab9_ct213$
```