Coin problem

Elizaveta Kostenok
MIPT, 2021

Task

To develop a model determining the total value of coins in the picture given

Assumptions

- The texture of table does not contain circles
- The optical axis of the camera is perpendicular to the table surface
- Coins can not overlap
- Coins can be located both tales and eagles
- Coins of all possible denominations can be in the picture

Strategy

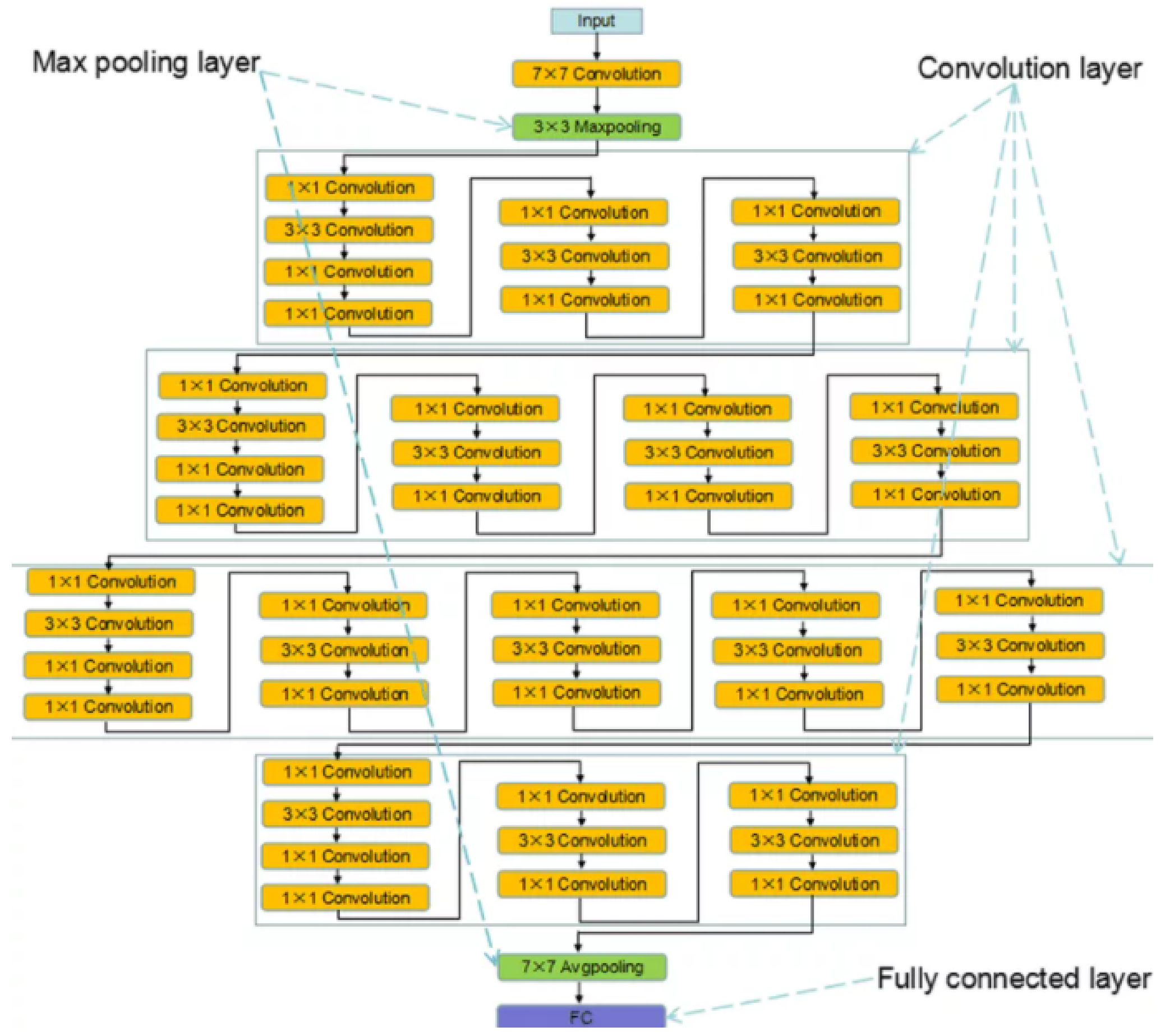
Circles detection + classification

Classification

My classifier is a convolutional neural network which extracts features from picture with the help of convolution operation and defines the class of the object. For creating an architecture of network and its training I used the *Transfer learning* method based on three stages:

- upload the state-of-the-art model pretrained on ImageNet dataset and freeze its weigths
- replace its classifier which is the last layer of model with new classifier for n classes where n is number of Simpsons
- train only weights of classifier

I used the ResNet50 architecture consisting of 50 dense layers, replaced and trained fully connected layer.

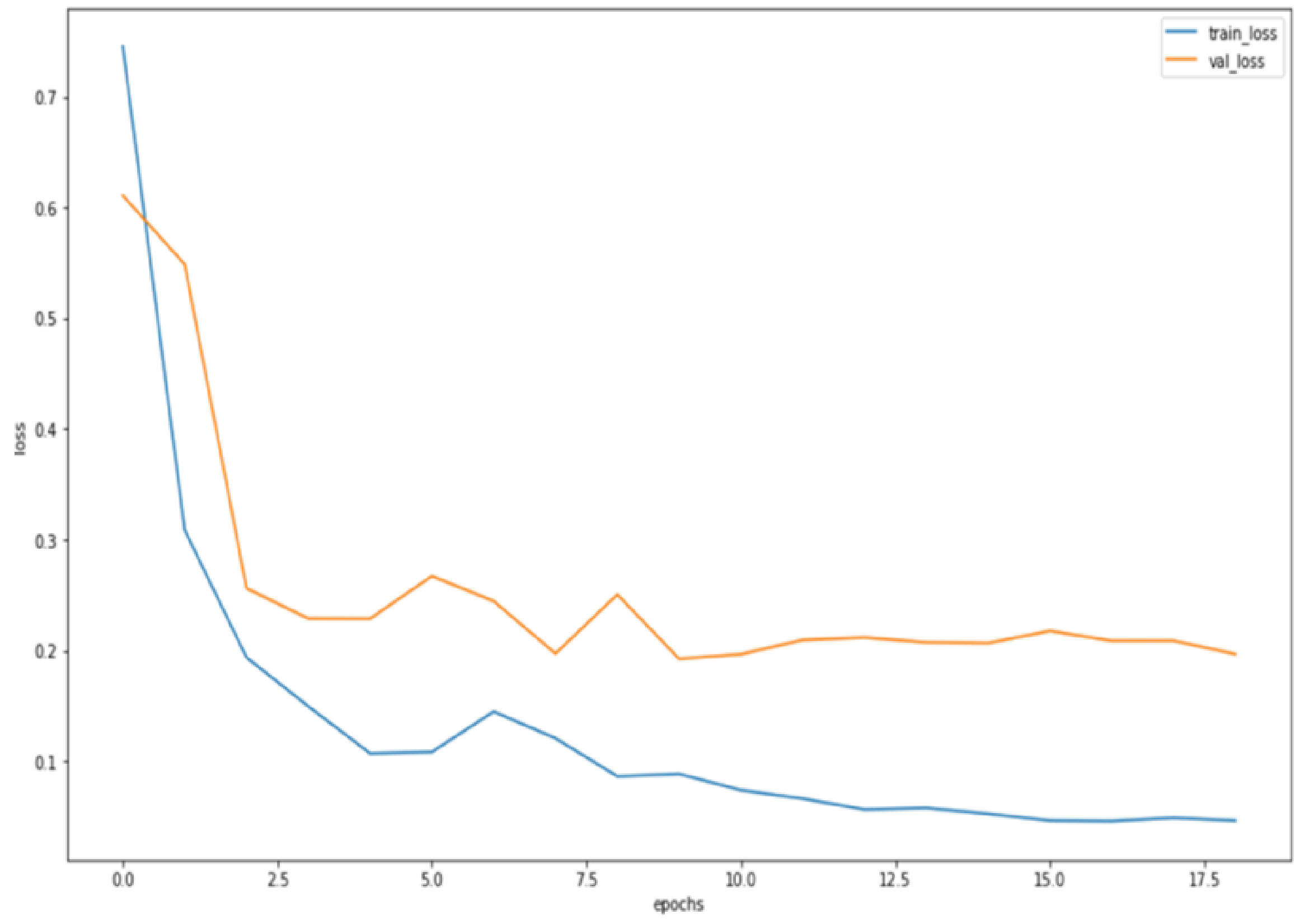


Training data

The dataset containing approximately 4000 pictures of Brazilian coins with right labels for training and validation was uploaded from kaggle.

Neural network training

I used the Adam with weight decay(AdamW) optimizer for training from torch.optim. According to the training graph, the number of epoch should not exceed 15 to prevent overfitting.



Quality metrics

The choice of the metric is due to the fact that the classes are balanced: some characters are secondary and they can be found in the train pictures less often than the main ones, so accuracy is not suitable. My task was a multi-class classification for 5 classes, for this reason I decided to analyze metrics precision and recall separately:

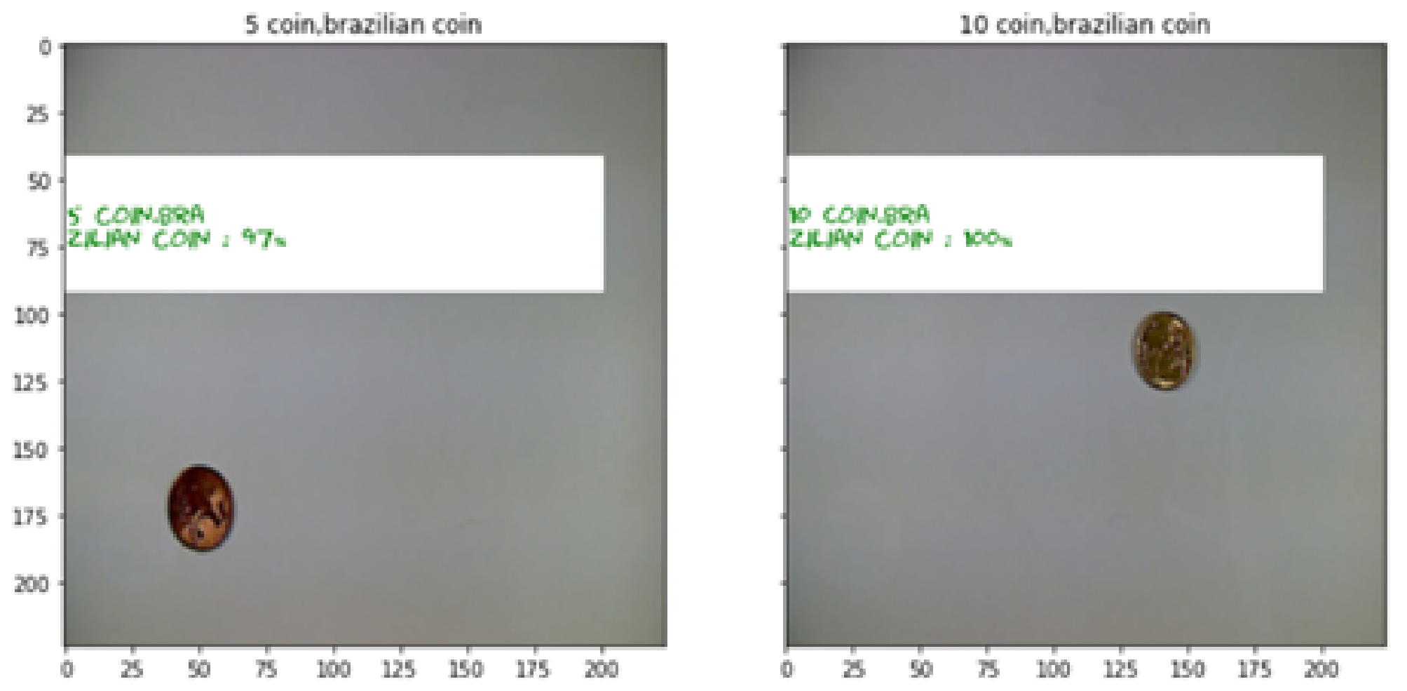
$$Precision(class = a) = \frac{TP(class{=}a)}{TP(class{=}a)+FP(class{=}a)}$$
$$Recall(class = a) = \frac{TP(class{=}a)}{TP(class{=}a)+FN(class{=}a)}$$

	Precision	Recall
5c	0,92	0,89
10c	0,98	0,96
25c	0,92	0,93
50c	0,9	0,94
100c	1	1

To estimate quality of model I used accuracy because classes are balanced in the dataset. It reached 0.98 for train data and 0.95 for test data.

Classification results

Visualization of model output is given below. True label is above each picture, predicted labels and confidence are in white rectangles.



Coins detection using OpenCV

Coins detection in my task is similar to circles detection in the picture due to assumptions mentioned before. I used the cv2.HoughCircles function from Open Source Computer Vision Library which detect edge pixels by the Canny edge detector and implements the Hough Transform algorithm. It is based on collecting evidence in an accumulator array for the presence of circles with different centers and radii and thresholding for selecting the most confident circles.

Detection results

The model was implemented which uses the cv2.HoughCircle function for finding coins in the picture and pretrained classifier for defining the value of each coin and displays a picture with detected coins. My model also predicts the total value summarising the most confident predictions for coins. The example of the coin detection is given: