FRUITS AND VEGETABLES RECOGNITION

Aleksander Świtała

CONVOLUTIONAL NEURAL NETWORK

Convolutional neural networks belong to deep neural networks and can gradually filter different parts of the training data and sharpen important features in the discrimination process used to recognize or classify patterns. This type of neural networks is predestined for computations on 2D structures.

The main part of these networks is a layer that uses an operation called convolution. Convolutions allow for the extraction of simple features in the initial layers of the network, e.g. they recognize edges with different orientations or stains of different colors, and then slices in subsequent layers.

NETWORK LAYERS

- Input Layer
- Convolutional Layer
- Pooling Layers
- Batch Normalizing Layer
- Flatten Layer

INPUT LAYER

Inputs images to the network and applies data normalization.

CONVOLUTIONAL LAYER

Convolutional layers contain learned filters that are used to distinguish images from each other. Parameters:

- Kernel/Filter Size - refers to the size that is moved over the window input.
- Padding - defines how the sample size is handled, which makes it possible to obtain the same size of the output as the size of the input.
- Strides - indicates by how many pixels the filter window should be moved.
- ReLU (Rectified Linear Activation) - uses non-linearity in the model. It is very quick to train.

POOLING LAYER

Uses down sampling on neighboring pixels with similar values to reduce size of the input, and as a result lowers the number of parameters to train, shortens the time of network working, and simplifies the model.

BATCH NORMALIZING LAYER

Normalization is the most effective way to fight network overtraining. It standardizes data making the process of training smoother, and faster.

FLATTEN LAYER

Flatten layer transforms the multidimensional network layer into a one-dimensional vector to match the input to the classification. The most commonly used function is SoftMax to classification of these features, which requires univariate data.

DATASET

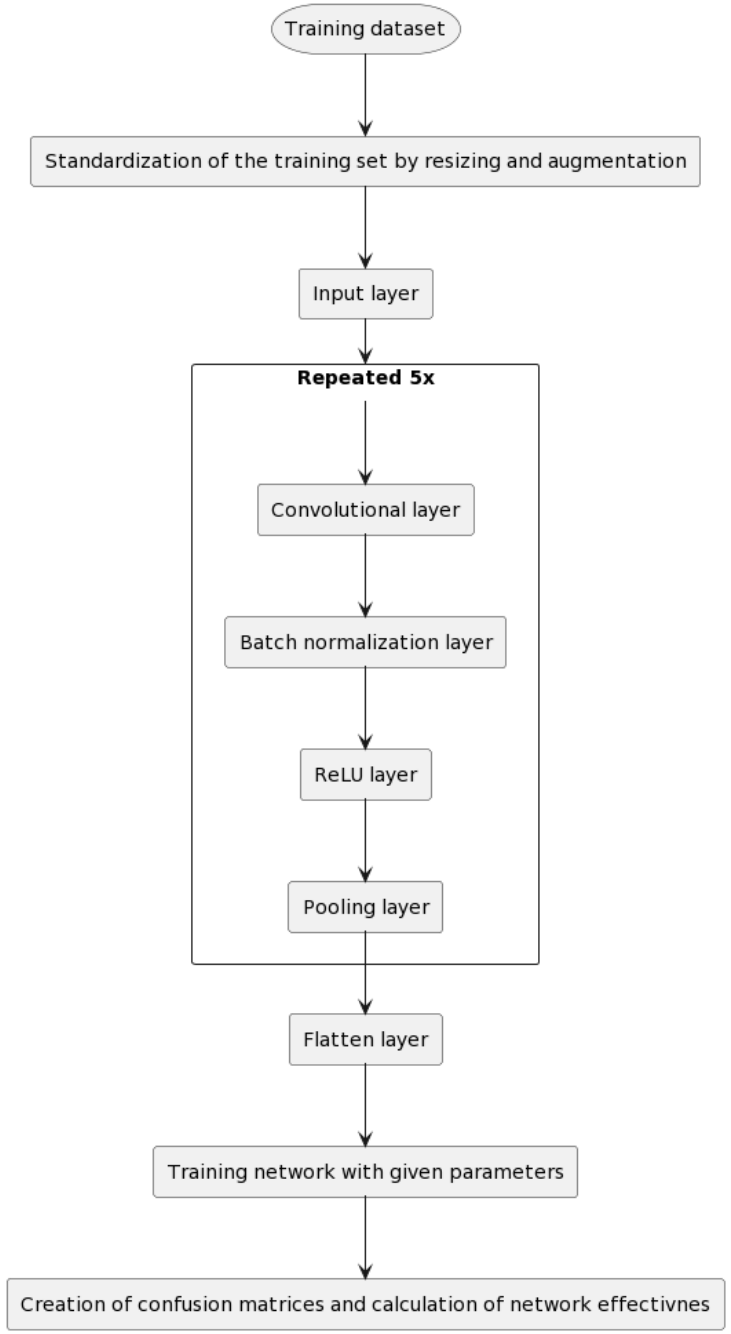
The used dataset was from kaggle website and consists of images of 30 types of fruits and vegetables bulk imported from Google Gallery.

There was a need to prepare the dataset before training, as there were different formats than .jpg, images of different dimensions, and channels size.

DATASET

	Train Set	Validation Set	Test set
Apple	499	49	50
Banana	540	65	65
Broccoli	496	50	50
Carrots	496	50	50
Cauliflower	498	50	50
Chili	395	50	50
Coconut	500	50	50
Cucumber	498	50	50
Custard Apple	500	50	50
Dates	496	50	50
Dragon	500	50	50
Egg	500	50	50
Garlic	500	50	50
Grape	499	50	50
Green Lemon	499	50	50
Jackfruit	500	50	50
Kiwi	497	50	50
Mango	500	50	50
Okra	500	50	50
Onion	500	50	50
Orange	496	48	48
Papaya	500	50	50
Peanut	500	50	50
Pineapple	500	50	50
Pomegranate	499	50	50
Star Fruit	496	50	50
Strawberry	500	50	50
Sweet Potato	499	50	50
Watermelon	497	50	50
White Mushrom	498	50	50
Together	14898	1512	1513

INTERNAL SPECS



TRAINED NETWORK PARAMS

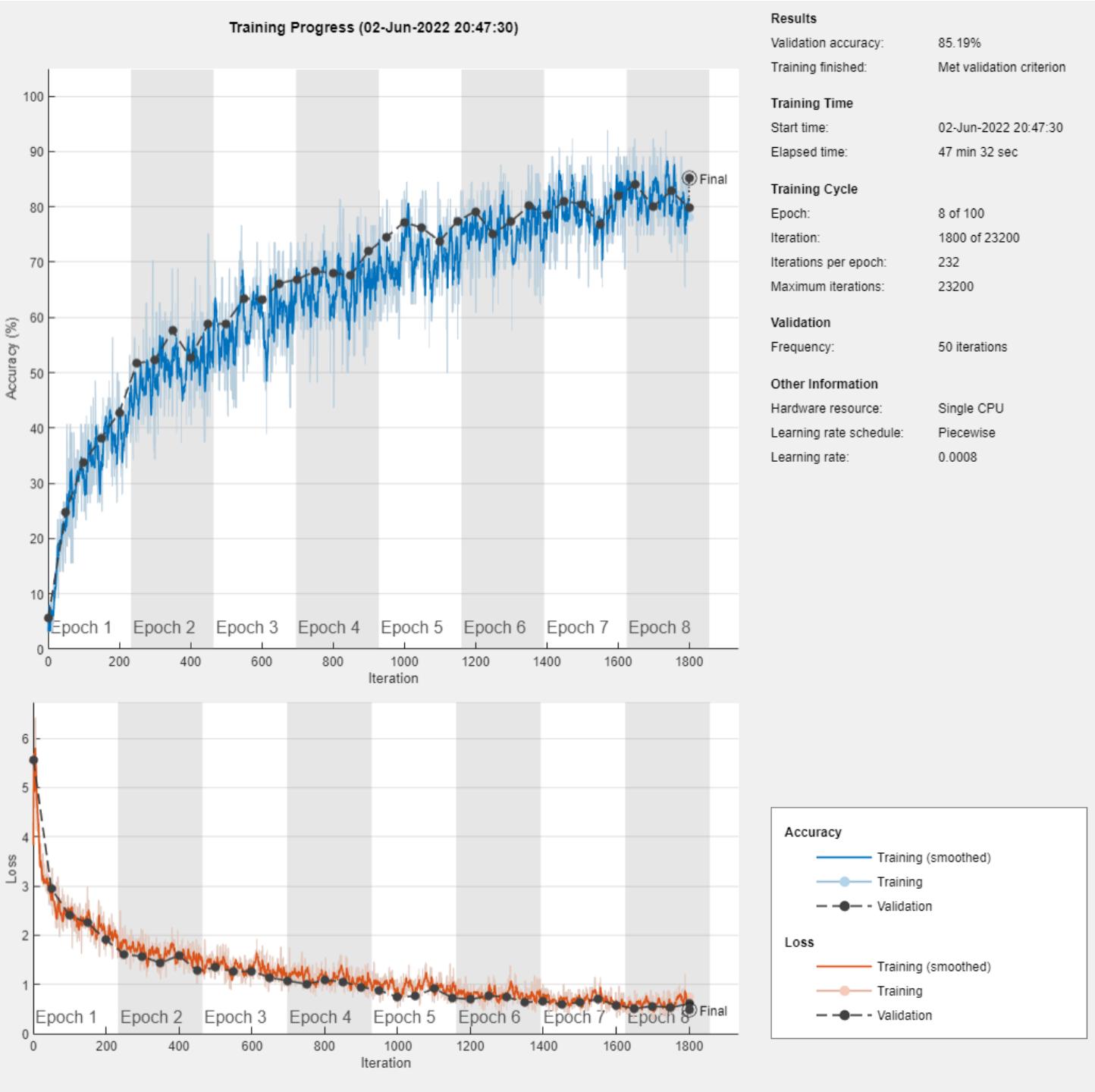
ImageDataAugmenter – RandRotation [-20,20], RandXTranslation [-3,3], RandYTranslation [-3,3]

Convolutional2dLater number of filters – 4/8/16/32/64

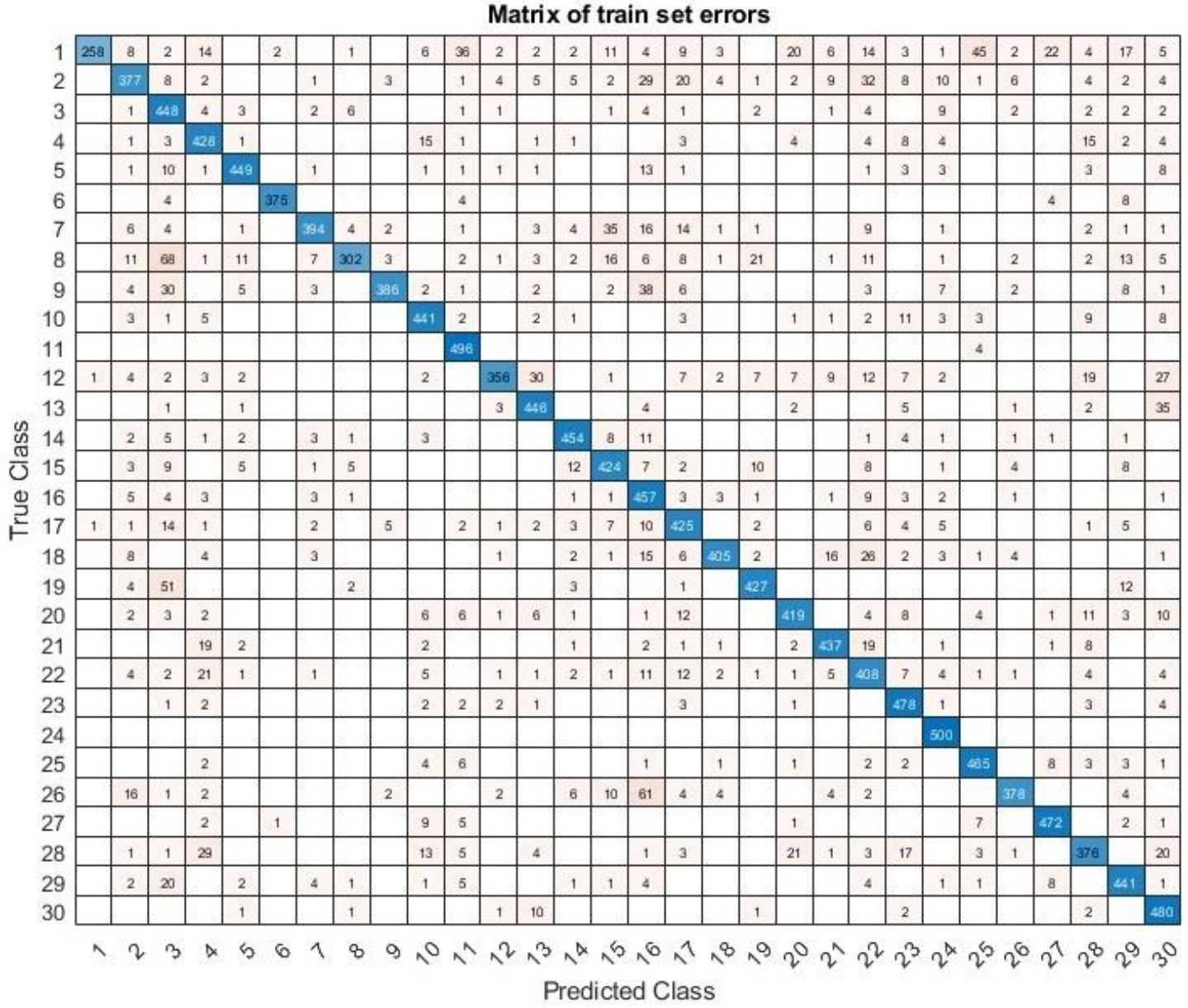
Convolutional2dLater size of filters – 3

TrainingOptions - SolverName adam, LearnRateDropFactor 0.8, LearnRateDropPeriod 5, ValidationPatience 3, OutputNetwork last-iteration

TRAINED NETWORK



TRAINED NETWORK



TRAINED NETWORK