Neural Network Training

Obliczenia inżynierskie w chmurze

Patryk Kruszynicki

# Image processing

The neural network, used to classify images, has been implemented in the Python programming language. The following libraries have been used:

-TensorFlow

-Keras

-OpenCV

-Numpy

-Random

-Os

The operation of the classification program based on neural networks consists in following the steps presented below.

- Loading input images.

- Rescaling images to size 128 x 128 pixels.

- Preparation of training data

Training data is saved as input\_data.npy and output\_data.cpy

# Neural network model

For the purpose of the classification problem, a convolutional network was chosen with the following structure.

- Input layer.

- 128 size convolution layer with ReLU activation function.

- Pooling layer covering 2 x 2 area.

- 128 size convolution layer with ReLU activation function.

- Pooling layer covering 2 x 2 area.

- A dense layer containing 128 neurons with ReLU activation function.

- Output layer containing 5 neurons with the softmax activation function.

The model of neural network is saved as model.h5 file.

# Master Docker

FROM python:3.6

RUN pip --no-cache-dir install keras

RUN pip install paramiko

ARG SLAVE\_IP\_0

ARG SLAVE\_PASSWORD\_0

ARG SLAVE\_IP\_1

ARG SLAVE\_PASSWORD\_1

ENV SLAVE\_IP0=${SLAVE\_IP\_0}

ENV SLAVE\_PASSWORD0=${SLAVE\_PASSWORD\_0}

ENV SLAVE\_IP1=${SLAVE\_IP\_1}

ENV SLAVE\_PASSWORD1=${SLAVE\_PASSWORD\_1}

COPY masterscript.py /masterscript.py

CMD python masterscript.py $SLAVE\_IP0 $SLAVE\_PASSWORD0 $SLAVE\_IP1 $SLAVE\_PASSWORD1

# Configuration of SSH

Port 2222

ListenAddress 0.0.0.0

LoginGraceTime 180

X11Forwarding yes

Ciphers aes256-ctr

MACs hmac-sha1

StrictModes yes

SyslogFacility DAEMON

PasswordAuthentication yes

PermitEmptyPasswords no

PermitRootLogin yes

Subsystem sftp internal-sftp

# Slave Docker

FROM python:3.6

RUN pip --no-cache-dir install keras

RUN pip --no-cache-dir install ssh

ARG var="Docker1!"

ENV SSH\_PASSWD "root:${var}"

RUN apt-get update \

&& apt-get install -y --no-install-recommends dialog \

&& apt-get update \

&& apt-get install -y --no-install-recommends openssh-server \

&& echo "$SSH\_PASSWD" | chpasswd

COPY sshd\_config /etc/ssh/

EXPOSE 2222

COPY input\_data.npy /input\_data.npy

COPY output\_data.npy /output\_data.npy

COPY slavescript.py /slavescript.py

RUN ["service", "ssh", "start"]

CMD ["/usr/sbin/sshd", "-D"]