Zadanie domowe: generator obrazów psów

Autorzy:

- Wojciech Rosenthal, indeks 180458
- Grzegorz Pozorski, indeks 180169

Informacje:

- Grupa: Uczenie maszynowe 2
- Przedmiot: Uczenie głębokie
- Data: 22.01.2024 r.
- Specjalność: Uczenie maszynowe

I pobranie danych

```
!curl https://download.microsoft.com/download/3/E/1/3E1C3F21-ECDB-
4869-8368-6DEBA77B919F/kagglecatsanddogs_5340.zip -o
kagglecatsanddogs_5340.zip

  % Total    % Received % Xferd  Average Speed   Time    Time     Time
Current
                                 Dload  Upload   Total   Spent    Left
Speed

  0     0    0     0    0     0      0      0 --:--:-- --:--:--
--:--:--     0
  0     0    0     0    0     0      0      0 --:--:-- --:--:--
--:--:--     0
  1  786M    1 8525k    0     0  7345k      0  0:01:49  0:00:01
0:01:48 7355k
  2  786M    2 17.7M    0     0  8376k      0  0:01:36  0:00:02
0:01:34 8385k
  3  786M    3 27.2M    0     0  8837k      0  0:01:31  0:00:03
0:01:28 8843k
  4  786M    4 36.9M    0     0  9084k      0  0:01:28  0:00:04
0:01:24 9089k
  5  786M    5 46.5M    0     0  9237k      0  0:01:27  0:00:05
0:01:22 9541k
  7  786M    7 56.2M    0     0  9341k      0  0:01:26  0:00:06
0:01:20 9804k
  8  786M    8 65.8M    0     0  9418k      0  0:01:25  0:00:07
0:01:18 9870k
  9  786M    9 75.2M    0     0  9443k      0  0:01:25  0:00:08
```

```
0:01:17 9825k
 10  786M   10 84.9M    0     0  9490k      0  0:01:24  0:00:09
0:01:15 9828k
 12  786M   12 94.5M    0     0  9525k      0  0:01:24  0:00:10
0:01:14 9822k
 12  786M   12  102M    0     0  9368k      0  0:01:25  0:00:11
0:01:14 9403k
 14  786M   14  111M    0     0  9406k      0  0:01:25  0:00:12
0:01:13 9390k
 15  786M   15  121M    0     0  9445k      0  0:01:25  0:00:13
0:01:12 9451k
 16  786M   16  131M    0     0  9476k      0  0:01:25  0:00:14
0:01:11 9451k
 17  786M   17  140M    0     0  9503k      0  0:01:24  0:00:15
0:01:09 9458k
 19  786M   19  150M    0     0  9526k      0  0:01:24  0:00:16
0:01:08 9876k
 20  786M   20  160M    0     0  9547k      0  0:01:24  0:00:17
0:01:07 9890k
 21  786M   21  169M    0     0  9565k      0  0:01:24  0:00:18
0:01:06 9881k
 22  786M   22  179M    0     0  9578k      0  0:01:24  0:00:19
0:01:05 9868k
 24  786M   24  188M    0     0  9593k      0  0:01:23  0:00:20
0:01:03 9867k
 24  786M   24  196M    0     0  9508k      0  0:01:24  0:00:21
0:01:03 9452k
 26  786M   26  206M    0     0  9524k      0  0:01:24  0:00:22
0:01:02 9446k
 27  786M   27  215M    0     0  9536k      0  0:01:24  0:00:23
0:01:01 9431k
 28  786M   28  225M    0     0  9550k      0  0:01:24  0:00:24
0:01:00 9441k
 29  786M   29  234M    0     0  9563k      0  0:01:24  0:00:25
0:00:59 9444k
 31  786M   31  244M    0     0  9573k      0  0:01:24  0:00:26
0:00:58 9845k
 32  786M   32  254M    0     0  9586k      0  0:01:24  0:00:27
0:00:57 9864k
 33  786M   33  263M    0     0  9597k      0  0:01:23  0:00:28
0:00:55 9875k
 34  786M   34  273M    0     0  9606k      0  0:01:23  0:00:29
0:00:54 9880k
 36  786M   36  283M    0     0  9615k      0  0:01:23  0:00:30
0:00:53 9876k
 37  786M   37  292M    0     0  9623k      0  0:01:23  0:00:31
0:00:52 9887k
 38  786M   38  302M    0     0  9630k      0  0:01:23  0:00:32
0:00:51 9869k
```

```
 39  786M   39  312M    0     0  9638k      0  0:01:23  0:00:33 
0:00:50 9870k
 40  786M   40  319M    0     0  9586k      0  0:01:24  0:00:34 
0:00:50 9467k
 41  786M   41  328M    0     0  9555k      0  0:01:24  0:00:35 
0:00:49 9196k
 42  786M   42  337M    0     0  9564k      0  0:01:24  0:00:36 
0:00:48 9194k
 44  786M   44  347M    0     0  9573k      0  0:01:24  0:00:37 
0:00:47 9203k
 45  786M   45  357M    0     0  9579k      0  0:01:24  0:00:38 
0:00:46 9190k
 46  786M   46  366M    0     0  9589k      0  0:01:24  0:00:39 
0:00:45 9604k
 47  786M   47  376M    0     0  9593k      0  0:01:23  0:00:40 
0:00:43 9852k
 49  786M   49  385M    0     0  9600k      0  0:01:23  0:00:41 
0:00:42 9855k
 50  786M   50  395M    0     0  9606k      0  0:01:23  0:00:42 
0:00:41 9853k
 51  786M   51  405M    0     0  9612k      0  0:01:23  0:00:43 
0:00:40 9867k
 52  786M   52  414M    0     0  9613k      0  0:01:23  0:00:44 
0:00:39 9804k
 53  786M   53  422M    0     0  9580k      0  0:01:24  0:00:45 
0:00:39 9478k
 54  786M   54  432M    0     0  9586k      0  0:01:24  0:00:46 
0:00:38 9470k
 56  786M   56  441M    0     0  9592k      0  0:01:23  0:00:47 
0:00:36 9473k
 57  786M   57  451M    0     0  9598k      0  0:01:23  0:00:48 
0:00:35 9476k
 58  786M   58  461M    0     0  9604k      0  0:01:23  0:00:49 
0:00:34 9525k
 59  786M   59  470M    0     0  9610k      0  0:01:23  0:00:50 
0:00:33 9881k
 61  786M   61  480M    0     0  9615k      0  0:01:23  0:00:51 
0:00:32 9887k
 62  786M   62  490M    0     0  9620k      0  0:01:23  0:00:52 
0:00:31 9883k
 63  786M   63  499M    0     0  9624k      0  0:01:23  0:00:53 
0:00:30 9876k
 64  786M   64  509M    0     0  9629k      0  0:01:23  0:00:54 
0:00:29 9874k
 65  786M   65  516M    0     0  9595k      0  0:01:23  0:00:55 
0:00:28 9439k
 66  786M   66  526M    0     0  9600k      0  0:01:23  0:00:56 
0:00:27 9443k
 68  786M   68  536M    0     0  9604k      0  0:01:23  0:00:57 
```

```
0:00:26 9443k
 69  786M   69  545M    0     0  9609k      0  0:01:23  0:00:58
0:00:25 9441k
 70  786M   70  555M    0     0  9612k      0  0:01:23  0:00:59
0:00:24 9420k
 71  786M   71  564M    0     0  9616k      0  0:01:23  0:01:00
0:00:23 9855k
 73  786M   73  574M    0     0  9620k      0  0:01:23  0:01:01
0:00:22 9852k
 74  786M   74  584M    0     0  9624k      0  0:01:23  0:01:02
0:00:21 9851k
 75  786M   75  593M    0     0  9629k      0  0:01:23  0:01:03
0:00:20 9860k
 76  786M   76  603M    0     0  9630k      0  0:01:23  0:01:04
0:00:19 9844k
 77  786M   77  610M    0     0  9601k      0  0:01:23  0:01:05
0:00:18 9422k
 78  786M   78  620M    0     0  9606k      0  0:01:23  0:01:06
0:00:17 9427k
 80  786M   80  630M    0     0  9610k      0  0:01:23  0:01:07
0:00:16 9425k
 81  786M   81  639M    0     0  9614k      0  0:01:23  0:01:08
0:00:15 9426k
 82  786M   82  649M    0     0  9618k      0  0:01:23  0:01:09
0:00:14 9463k
 83  786M   83  659M    0     0  9620k      0  0:01:23  0:01:10
0:00:13 9869k
 85  786M   85  668M    0     0  9624k      0  0:01:23  0:01:11
0:00:12 9872k
 86  786M   86  678M    0     0  9628k      0  0:01:23  0:01:12
0:00:11 9873k
 87  786M   87  688M    0     0  9631k      0  0:01:23  0:01:13
0:00:10 9872k
 88  786M   88  697M    0     0  9635k      0  0:01:23  0:01:14
0:00:09 9871k
 89  786M   89  707M    0     0  9638k      0  0:01:23  0:01:15
0:00:08 9881k
 91  786M   91  717M    0     0  9640k      0  0:01:23  0:01:16
0:00:07 9867k
 92  786M   92  726M    0     0  9644k      0  0:01:23  0:01:17
0:00:06 9872k
 93  786M   93  736M    0     0  9647k      0  0:01:23  0:01:18
0:00:05 9870k
 94  786M   94  743M    0     0  9615k      0  0:01:23  0:01:19
0:00:04 9324k
 95  786M   95  752M    0     0  9618k      0  0:01:23  0:01:20
0:00:03 9328k
 96  786M   96  762M    0     0  9622k      0  0:01:23  0:01:21
0:00:02 9341k
```

```
 98  786M   98  772M    0     0  9625k      0  0:01:23  0:01:22
0:00:01 9340k
 99  786M   99  781M    0     0  9628k      0  0:01:23  0:01:23
--:--:-- 9327k
100  786M  100  786M    0     0  9629k      0  0:01:23  0:01:23
--:--:-- 9876k
```

```
!unzip kagglecatsanddogs_5340.zip
```

```
'unzip' is not recognized as an internal or external command,
operable program or batch file.
```

II przetwarzanie wstępne

```
import numpy as np
import tensorflow as tf
import cv2
import time
import torch
from PIL import Image
from IPython.display import display
import matplotlib.pyplot as plt
import tensorflow.keras.backend as K

import os
import random

print(tf.__version__)
```

```
WARNING:tensorflow:From C:\Users\grzeg\PycharmProjects\pg-workshop\UG\
vae-ug-homework\.venv\lib\site-packages\keras\src\losses.py:2976: The
name tf.losses.sparse_softmax_cross_entropy is deprecated. Please use
tf.compat.v1.losses.sparse_softmax_cross_entropy instead.
2.15.0
```

```
IMAGE_WIDTH=64
IMAGE_HEIGHT=64
IMAGE_SIZE=(IMAGE_WIDTH, IMAGE_HEIGHT)
IMAGE_CHANNELS=3
```

```
dirpath = 'kagglecatsanddogs_5340/PetImages/Dog/'
filenames = [filename for filename in os.listdir(dirpath) if
filename.endswith(".jpg")]
print(f"Total number of images: {len(filenames)}")
```

```
Total number of images: 12500
```

```
sample = dirpath + random.choice(filenames)
img = Image.open(sample)
display(img)
```

Wykrycie psów na obrazach i przycięcie ich

```
def make_square(image):
    height, width = image.shape[:2]

    max_side = max(height, width)
    top = bottom = left = right = 0

    if height < max_side:
        diff = max_side - height
        top = diff // 2
        bottom = diff - top
    elif width < max_side:
        diff = max_side - width
        left = diff // 2
        right = diff - left

    mean_color = cv2.mean(image)
    mean_color = tuple(map(int, mean_color))
    padded_image = cv2.copyMakeBorder(image, top, bottom, left, right,
cv2.BORDER_CONSTANT, value=mean_color)

    return padded_image

preprocessed_dirpath = 'kagglecatsanddogs_5340/PetImages/Dog-
preprocessed/'

if not os.path.exists(preprocessed_dirpath):
    os.makedirs(preprocessed_dirpath)
```

```python
model = torch.hub.load('ultralytics/yolov5', 'yolov5s',
force_reload=True, trust_repo=True)
```

```
Downloading: "https://github.com/ultralytics/yolov5/zipball/master" to
/home/wojtero/.cache/torch/hub/master.zip

Collecting ultralytics
  Downloading ultralytics-8.1.3-py3-none-any.whl.metadata (40 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 40.2/40.2 kB 1.0 MB/s eta
0:00:00
atplotlib>=3.3.0 (from ultralytics)
  Downloading matplotlib-3.8.2-cp310-cp310-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (5.8 kB)
Requirement already satisfied: numpy>=1.22.2 in
./venv/lib/python3.10/site-packages (from ultralytics) (1.26.3)
Requirement already satisfied: opencv-python>=4.6.0 in
./venv/lib/python3.10/site-packages (from ultralytics) (4.9.0.80)
Requirement already satisfied: pillow>=7.1.2 in
./venv/lib/python3.10/site-packages (from ultralytics) (10.2.0)
Requirement already satisfied: pyyaml>=5.3.1 in
./venv/lib/python3.10/site-packages (from ultralytics) (6.0.1)
Requirement already satisfied: requests>=2.23.0 in
./venv/lib/python3.10/site-packages (from ultralytics) (2.31.0)
Collecting scipy>=1.4.1 (from ultralytics)
  Downloading scipy-1.11.4-cp310-cp310-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (60 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 60.4/60.4 kB 1.9 MB/s eta
0:00:00
ent already satisfied: torch>=1.8.0 in ./venv/lib/python3.10/site-
packages (from ultralytics) (2.1.2)
Collecting torchvision>=0.9.0 (from ultralytics)
  Downloading torchvision-0.16.2-cp310-cp310-
manylinux1_x86_64.whl.metadata (6.6 kB)
Collecting tqdm>=4.64.0 (from ultralytics)
  Downloading tqdm-4.66.1-py3-none-any.whl.metadata (57 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 57.6/57.6 kB 4.0 MB/s eta
0:00:00
ent already satisfied: psutil in ./venv/lib/python3.10/site-packages
(from ultralytics) (5.9.8)
Collecting py-cpuinfo (from ultralytics)
  Downloading py_cpuinfo-9.0.0-py3-none-any.whl (22 kB)
Collecting thop>=0.1.1 (from ultralytics)
  Downloading thop-0.1.1.post2209072238-py3-none-any.whl (15 kB)
Requirement already satisfied: pandas>=1.1.4 in
./venv/lib/python3.10/site-packages (from ultralytics) (2.2.0)
Collecting seaborn>=0.11.0 (from ultralytics)
  Downloading seaborn-0.13.1-py3-none-any.whl.metadata (5.4 kB)
Collecting contourpy>=1.0.1 (from matplotlib>=3.3.0->ultralytics)
  Downloading contourpy-1.2.0-cp310-cp310-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (5.8 kB)
```

```
Collecting cycler>=0.10 (from matplotlib>=3.3.0->ultralytics)
  Downloading cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)
Collecting fonttools>=4.22.0 (from matplotlib>=3.3.0->ultralytics)
  Downloading fonttools-4.47.2-cp310-cp310-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (157 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 157.6/157.6 kB 2.1 MB/s eta
0:00:0000:0100:01
 matplotlib>=3.3.0->ultralytics)
  Downloading kiwisolver-1.4.5-cp310-cp310-
manylinux_2_12_x86_64.manylinux2010_x86_64.whl.metadata (6.4 kB)
Requirement already satisfied: packaging>=20.0 in
./venv/lib/python3.10/site-packages (from matplotlib>=3.3.0-
>ultralytics) (23.2)
Collecting pyparsing>=2.3.1 (from matplotlib>=3.3.0->ultralytics)
  Downloading pyparsing-3.1.1-py3-none-any.whl.metadata (5.1 kB)
Requirement already satisfied: python-dateutil>=2.7 in
./venv/lib/python3.10/site-packages (from matplotlib>=3.3.0-
>ultralytics) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in
./venv/lib/python3.10/site-packages (from pandas>=1.1.4->ultralytics)
(2023.3.post1)
Requirement already satisfied: tzdata>=2022.7 in
./venv/lib/python3.10/site-packages (from pandas>=1.1.4->ultralytics)
(2023.4)
Requirement already satisfied: charset-normalizer<4,>=2 in
./venv/lib/python3.10/site-packages (from requests>=2.23.0-
>ultralytics) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in
./venv/lib/python3.10/site-packages (from requests>=2.23.0-
>ultralytics) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in
./venv/lib/python3.10/site-packages (from requests>=2.23.0-
>ultralytics) (2.1.0)
Requirement already satisfied: certifi>=2017.4.17 in
./venv/lib/python3.10/site-packages (from requests>=2.23.0-
>ultralytics) (2023.11.17)
Requirement already satisfied: filelock in ./venv/lib/python3.10/site-
packages (from torch>=1.8.0->ultralytics) (3.13.1)
Requirement already satisfied: typing-extensions in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(4.9.0)
Requirement already satisfied: sympy in ./venv/lib/python3.10/site-
packages (from torch>=1.8.0->ultralytics) (1.12)
Requirement already satisfied: networkx in ./venv/lib/python3.10/site-
packages (from torch>=1.8.0->ultralytics) (3.2.1)
Requirement already satisfied: jinja2 in ./venv/lib/python3.10/site-
packages (from torch>=1.8.0->ultralytics) (3.1.3)
Requirement already satisfied: fsspec in ./venv/lib/python3.10/site-
packages (from torch>=1.8.0->ultralytics) (2023.12.2)
```

```
Requirement already satisfied: nvidia-cuda-nvrtc-cu12==12.1.105 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(12.1.105)
Requirement already satisfied: nvidia-cuda-runtime-cu12==12.1.105
in ./venv/lib/python3.10/site-packages (from torch>=1.8.0-
>ultralytics) (12.1.105)
Requirement already satisfied: nvidia-cuda-cupti-cu12==12.1.105 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(12.1.105)
Requirement already satisfied: nvidia-cudnn-cu12==8.9.2.26 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(8.9.2.26)
Requirement already satisfied: nvidia-cublas-cu12==12.1.3.1 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(12.1.3.1)
Requirement already satisfied: nvidia-cufft-cu12==11.0.2.54 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(11.0.2.54)
Requirement already satisfied: nvidia-curand-cu12==10.3.2.106 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(10.3.2.106)
Requirement already satisfied: nvidia-cusolver-cu12==11.4.5.107 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(11.4.5.107)
Requirement already satisfied: nvidia-cusparse-cu12==12.1.0.106 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(12.1.0.106)
Requirement already satisfied: nvidia-nccl-cu12==2.18.1 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(2.18.1)
Requirement already satisfied: nvidia-nvtx-cu12==12.1.105 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(12.1.105)
Requirement already satisfied: triton==2.1.0 in
./venv/lib/python3.10/site-packages (from torch>=1.8.0->ultralytics)
(2.1.0)
Requirement already satisfied: nvidia-nvjitlink-cu12 in
./venv/lib/python3.10/site-packages (from nvidia-cusolver-
cu12==11.4.5.107->torch>=1.8.0->ultralytics) (12.3.101)
Requirement already satisfied: six>=1.5 in ./venv/lib/python3.10/site-
packages (from python-dateutil>=2.7->matplotlib>=3.3.0->ultralytics)
(1.16.0)
Requirement already satisfied: MarkupSafe>=2.0 in
./venv/lib/python3.10/site-packages (from jinja2->torch>=1.8.0-
>ultralytics) (2.1.4)
Requirement already satisfied: mpmath>=0.19 in
./venv/lib/python3.10/site-packages (from sympy->torch>=1.8.0-
>ultralytics) (1.3.0)
Downloading ultralytics-8.1.3-py3-none-any.whl (701 kB)
```

```
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 701.1/701.1 kB 7.0 MB/s eta
0:00:00a 0:00:01m
atplotlib-3.8.2-cp310-cp310-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.6 MB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11.6/11.6 MB 7.2 MB/s eta
0:00:0000:0100:01
anylinux_2_17_x86_64.manylinux2014_x86_64.whl (36.4 MB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 36.4/36.4 MB 34.6 MB/s eta
0:00:0000:0100:01
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 294.8/294.8 kB 40.7 MB/s eta
0:00:00
anylinux1_x86_64.whl (6.8 MB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.8/6.8 MB 40.4 MB/s eta
0:00:00a 0:00:01
-4.66.1-py3-none-any.whl (78 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 78.3/78.3 kB 27.2 MB/s eta
0:00:00
anylinux_2_17_x86_64.manylinux2014_x86_64.whl (310 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 310.7/310.7 kB 28.7 MB/s eta
0:00:00
anylinux_2_17_x86_64.manylinux2014_x86_64.whl (4.6 MB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 4.6/4.6 MB 34.8 MB/s eta
0:00:00a 0:00:01
anylinux_2_12_x86_64.manylinux2010_x86_64.whl (1.6 MB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.6/1.6 MB 38.3 MB/s eta
0:00:0000:01
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 103.1/103.1 kB 24.3 MB/s eta
0:00:00
, scipy, pyparsing, kiwisolver, fonttools, cycler, contourpy,
matplotlib, seaborn, torchvision, thop, ultralytics
Successfully installed contourpy-1.2.0 cycler-0.12.1 fonttools-4.47.2
kiwisolver-1.4.5 matplotlib-3.8.2 py-cpuinfo-9.0.0 pyparsing-3.1.1
scipy-1.11.4 seaborn-0.13.1 thop-0.1.1.post2209072238 torchvision-
0.16.2 tqdm-4.66.1 ultralytics-8.1.3
requirements: Ultralytics requirement ['gitpython>=3.1.30'] not found,
attempting AutoUpdate...
Collecting gitpython>=3.1.30
  Downloading GitPython-3.1.41-py3-none-any.whl.metadata (14 kB)
Collecting gitdb<5,>=4.0.1 (from gitpython>=3.1.30)
  Downloading gitdb-4.0.11-py3-none-any.whl.metadata (1.2 kB)
Collecting smmap<6,>=3.0.1 (from gitdb<5,>=4.0.1->gitpython>=3.1.30)
  Downloading smmap-5.0.1-py3-none-any.whl.metadata (4.3 kB)
Downloading GitPython-3.1.41-py3-none-any.whl (196 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 196.4/196.4 kB 1.5 MB/s eta
0:00:00 0:00:010m
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 62.7/62.7 kB 8.0 MB/s eta
0:00:00
map-5.0.1-py3-none-any.whl (24 kB)
Installing collected packages: smmap, gitdb, gitpython
```

```
Successfully installed gitdb-4.0.11 gitpython-3.1.41 smmap-5.0.1

requirements: AutoUpdate success ✅ 2.0s, installed 1 package:
['gitpython>=3.1.30']
requirements: ⚠️ Restart runtime or rerun command for updates to take
effect

YOLOv5 🚀 2024-1-20 Python-3.10.12 torch-2.1.2+cu121 CUDA:0 (NVIDIA
GeForce RTX 4080, 16057MiB)

Downloading
https://github.com/ultralytics/yolov5/releases/download/v7.0/yolov5s.p
t to yolov5s.pt...
100.0%

Fusing layers...
YOLOv5s summary: 213 layers, 7225885 parameters, 0 gradients, 16.4
GFLOPs
Adding AutoShape...
```

```
sample = 'kagglecatsanddogs_5340/PetImages/Dog/' + '10003.jpg'
image = cv2.imread(sample, cv2.IMREAD_COLOR)
results = model(image)

pd_results = results.pandas().xyxy[0]
pd_results = pd_results[pd_results['name'] == 'dog']
pd_results = pd_results[pd_results['confidence'] > 0.5]
top, bottom, left, right = pd_results[['ymax', 'ymin', 'xmin',
'xmax']].values[0]
left, right, top, bottom = round(left), round(right), round(top),
round(bottom)

image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
cropped_image = image[bottom:top, left:right]

img = Image.fromarray(cropped_image)
display(img)
```

```python
def process_image(filename, dirpath, preprocessed_dirpath):
    try:
        image = cv2.imread(dirpath + filename, cv2.IMREAD_COLOR)
        width, height = image.shape[:2]

        results = model(image)

        pd_results = results.pandas().xyxy[0]
        pd_results = pd_results[pd_results['name'] == 'dog']
        pd_results = pd_results[pd_results['confidence'] > 0.5]
        if len(pd_results) == 0:
            return None
        top, bottom, left, right = pd_results[['ymax', 'ymin', 'xmin',
'xmax']].values[0]
        left, right, top, bottom = round(left), round(right),
round(top), round(bottom)

        image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        cropped_image = image[bottom:top, left:right]

        padded_image = make_square(cropped_image)
        padded_resized_image = cv2.resize(padded_image, IMAGE_SIZE)
        cv2.imwrite(preprocessed_dirpath + filename,
padded_resized_image)

        return width, height
    except Exception as e:
        raise Exception(f"Error processing {filename}: {e}")

mean_size = np.zeros(2)
cnt = 0

start_time = time.time()
for filename in filenames:
    try:
        shape = process_image(filename, dirpath, preprocessed_dirpath)
        if shape:
            mean_size += np.array(shape[:2])
            cnt += 1
    except Exception as e:
        print(f"An error occurred: {e}")

end_time = time.time()
print(f"Total time: {end_time - start_time}")
print(f"Mean size: {mean_size / cnt}.\nTotal count: {cnt}")
```

```
An error occurred: Error processing 7112.jpg: 'NoneType' object has no
attribute 'shape'
An error occurred: Error processing 6059.jpg: 'NoneType' object has no
attribute 'shape'

Corrupt JPEG data: 1403 extraneous bytes before marker 0xd9

An error occurred: Error processing 7459.jpg: 'NoneType' object has no
attribute 'shape'

Corrupt JPEG data: 254 extraneous bytes before marker 0xd9

An error occurred: Error processing 10401.jpg: 'NoneType' object has
no attribute 'shape'
An error occurred: Error processing 10158.jpg: 'NoneType' object has
no attribute 'shape'

Corrupt JPEG data: 226 extraneous bytes before marker 0xd9

An error occurred: Error processing 1866.jpg: 'NoneType' object has no
attribute 'shape'
An error occurred: Error processing 10797.jpg: 'NoneType' object has
no attribute 'shape'

Corrupt JPEG data: 65 extraneous bytes before marker 0xd9

An error occurred: Error processing 3136.jpg: 'NoneType' object has no
attribute 'shape'
```

```
An error occurred: Error processing 4367.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 10747.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 11849.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 6718.jpg: 'NoneType' object has no attribute 'shape'

Corrupt JPEG data: 399 extraneous bytes before marker 0xd9

An error occurred: Error processing 5736.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 7133.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 9188.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 5604.jpg: 'NoneType' object has no attribute 'shape'

Corrupt JPEG data: 162 extraneous bytes before marker 0xd9
Warning: unknown JFIF revision number 0.00

An error occurred: Error processing 3288.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 6238.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 8730.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 3588.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 11702.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 7969.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 1308.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 7369.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 2688.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 2384.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 11853.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 11410.jpg: 'NoneType' object has no attribute 'shape'

Corrupt JPEG data: 2230 extraneous bytes before marker 0xd9
```

```
An error occurred: Error processing 11675.jpg: 'NoneType' object has no attribute 'shape'
An error occurred: Error processing 2877.jpg: 'NoneType' object has no attribute 'shape'
Total time: 72.31985402107239
Mean size: [     380.97      420.61].
Total count: 6489
```

Total time: 1510.6200618743896

Mean size: [381 420.6]

Total count: 6487

Wczytanie przetworzonego zbioru danych

```
preprocessed_dirpath = 'kagglecatsanddogs_5340/PetImages/Dog-preprocessed/'
filenames = [filename for filename in os.listdir(preprocessed_dirpath)]
print(f"Total number of images: {len(filenames)}")
```

```
Total number of images: 6487
```

```
X = np.zeros((len(filenames), IMAGE_HEIGHT, IMAGE_WIDTH, 1), dtype=np.uint8)
for index, filename in enumerate(filenames):
    image = cv2.imread(preprocessed_dirpath + filename, cv2.IMREAD_GRAYSCALE)
    X[index] = image.reshape(IMAGE_HEIGHT, IMAGE_WIDTH, 1)

X = X / 255.0
X_orig = X.copy()

print(X.shape)
```

```
(6487, 64, 64, 1)
```

```
ratio = 0.8

X = X_orig.copy()

split_index = round(X.shape[0] * ratio)
X_train_val = X[:split_index]
X_test = X[split_index:]

split_index_train_val = round(X_train_val.shape[0] * ratio)
X_train = X_train_val[:split_index_train_val]
X_val = X_train_val[split_index_train_val:]

print("Training set shape:", X_train.shape)
print("Validation set shape:", X_val.shape)
print("Test set shape:", X_test.shape)
```

```
Training set shape: (4152, 64, 64, 1)
Validation set shape: (1038, 64, 64, 1)
Test set shape: (1297, 64, 64, 1)
```

```
train_size = X_train.shape[0]
val_size = X_val.shape[0]
test_size = X_test.shape[0]
```

III Głęboki autokoder wariacyjny

```
# zainspirowane wykładem
# inspired by
https://github.com/shashankdhar/VAE-MNIST/blob/master/VAE.py
def latent_reparam_sampling(args):
    z_mean_layer, z_log_var_layer = args
    batch_size = K.shape(z_mean_layer)[0]
    data_size = K.int_shape(z_mean_layer)[1]
    epsilon = K.random_normal(shape=(batch_size, data_size))
    return z_mean_layer + K.exp(0.5 * z_log_var_layer) * epsilon

class DeepVae(tf.keras.Model):
    def __init__(self, latent_dimension):
        super(DeepVae, self).__init__()

        self.latent_dimension = latent_dimension

        self.encoder = tf.keras.Sequential([
            tf.keras.layers.Input(shape=(64, 64, 1)),
            tf.keras.layers.Conv2D(filters=32, kernel_size=3,
strides=(2, 2), activation='relu'),
            tf.keras.layers.Conv2D(filters=64, kernel_size=3,
strides=(2, 2), activation='relu'),
            tf.keras.layers.Flatten(),
            tf.keras.layers.Dense(self.latent_dimension +
self.latent_dimension),
        ], name='encoder')

        self.decoder = tf.keras.Sequential([
            tf.keras.layers.Input(shape=(self.latent_dimension,)),
            tf.keras.layers.Dense(units=16*16*32,
activation=tf.nn.relu),
            tf.keras.layers.Reshape(target_shape=(16, 16, 32)),
            tf.keras.layers.Conv2DTranspose(filters=64, kernel_size=3,
strides=2, padding='same', activation='relu'),
            tf.keras.layers.Conv2DTranspose(filters=32, kernel_size=3,
strides=2, padding='same', activation='relu'),
            tf.keras.layers.Conv2DTranspose(filters=1, kernel_size=3, 
strides=1, padding='same'),
        ], name='decoder')

    def get_z_mean_and_z_log_var(self, x):
        z_mean, z_logvar = tf.split(self.encoder(x), 
num_or_size_splits=2, axis=1)
        return z_mean, z_logvar

    def save(self, path):
        self.encoder.save_weights(f'{path}_encoder.h5')
        self.decoder.save_weights(f'{path}_decoder.h5')

    def load(self, path):
        self.encoder.load_weights(f'{path}_encoder.h5')
        self.decoder.load_weights(f'{path}_decoder.h5')

def compute_loss(model, x, beta=1.0):
    z_mean, z_log_var = model.get_z_mean_and_z_log_var(x)
    x_p = model.decoder(latent_reparam_sampling((z_mean, z_log_var)))
    loss_reconstruction = K.sum(K.binary_crossentropy(x, x_p), axis=-
1)
    loss_KL = 1 + z_log_var - K.square(z_mean) - K.exp(z_log_var)
    loss_KL = -0.5 * K.sum(loss_KL, axis=-1)
    loss = K.mean(loss_reconstruction) + beta * loss_KL
    return loss, loss_reconstruction, loss_KL

@tf.function
def train_step(model, x, optimizer, beta=1.0):
    with tf.GradientTape() as tape:
        loss, _, _ = compute_loss(model, x, beta)
    gradients = tape.gradient(loss, model.trainable_variables)
    optimizer.apply_gradients(zip(gradients, 
model.trainable_variables))
    return loss

epochs = 120
annealing_cycles = 4
latent_dimension = 1024

model = DeepVae(latent_dimension)
model.encoder.summary()
model.decoder.summary()
```

```
WARNING:tensorflow:From C:\Users\grzeg\PycharmProjects\pg-workshop\UG\
vae-ug-homework\.venv\lib\site-packages\keras\src\backend.py:873: The 
name tf.get_default_graph is deprecated. Please use 
tf.compat.v1.get_default_graph instead.
Model: "encoder"

_________________________________________________________________
```

```
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 31, 31, 32)        320

 conv2d_1 (Conv2D)           (None, 15, 15, 64)        18496

 flatten (Flatten)           (None, 14400)             0

 dense (Dense)               (None, 2048)              29493248

=================================================================
Total params: 29512064 (112.58 MB)
Trainable params: 29512064 (112.58 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
Model: "decoder"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_1 (Dense)             (None, 8192)              8396800

 reshape (Reshape)           (None, 16, 16, 32)        0

 conv2d_transpose (Conv2DTr  (None, 32, 32, 64)        18496
 anspose)

 conv2d_transpose_1 (Conv2D  (None, 64, 64, 32)        18464
 Transpose)

 conv2d_transpose_2 (Conv2D  (None, 64, 64, 1)         289
 Transpose)

=================================================================
Total params: 8434049 (32.17 MB)
Trainable params: 8434049 (32.17 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

```
optimizer = tf.keras.optimizers.Adam(1e-4)
batch_size = 128

train_dataset =
(tf.data.Dataset.from_tensor_slices(X_train.astype(np.float32)).shuffl
e(train_size).batch(batch_size))
val_dataset =
(tf.data.Dataset.from_tensor_slices(X_val.astype(np.float32)).shuffle(
val_size).batch(batch_size))
test_dataset =
(tf.data.Dataset.from_tensor_slices(X_test.astype(np.float32)).shuffle
(test_size).batch(batch_size))
```

```
annealing_cycle_length = epochs // annealing_cycles
best_loss = None

for epoch in range(1, epochs + 1):
    start_time = time.time()
    beta = min(float(int(epoch) % int(annealing_cycle_length)) /
annealing_cycle_length, 1.0)
    train_loss = tf.keras.metrics.Mean()
    val_loss = tf.keras.metrics.Mean()
    for train_x in train_dataset:
        train_loss(train_step(model, train_x, optimizer, beta))
    for val_x in val_dataset:
        val_loss(compute_loss(model, val_x)[0])
    end_time = time.time()

    if best_loss is None or train_loss.result() < best_loss:
        best_loss = train_loss.result()
        model.save('best')

    print(f'Epoch {epoch}: train_loss={train_loss.result()},
val_loss={val_loss.result()}, beta={beta}, time={end_time -
start_time}')
```

```
Epoch 1: train_loss=2.1104588508605957, val_loss=1.4946001768112183,
beta=0.03333333333333333, time=22.639453649520874
Epoch 2: train_loss=1.0628458261489868, val_loss=1.03101646900177,
beta=0.06666666666666667, time=17.791181087493896
Epoch 3: train_loss=0.939510703086853, val_loss=0.9175639748573303,
beta=0.1, time=18.08599328994751
Epoch 4: train_loss=0.8565689325332642, val_loss=0.8327543139457703,
beta=0.13333333333333333, time=18.602849006652832
Epoch 5: train_loss=0.7947717905044556, val_loss=0.7727011442184448,
beta=0.16666666666666666, time=17.969401121139526
Epoch 6: train_loss=0.7531177401542664, val_loss=0.7386229038238525,
beta=0.2, time=17.85089612007141
Epoch 7: train_loss=0.7295197248458862, val_loss=0.7219509482383728,
beta=0.23333333333333334, time=18.16361904144287
Epoch 8: train_loss=0.71897953748703, val_loss=0.715250551700592,
beta=0.26666666666666666, time=18.21601128578186
Epoch 9: train_loss=0.714522123336792, val_loss=0.7125479578971863,
beta=0.3, time=18.423256635665894
Epoch 10: train_loss=0.7127305269241333, val_loss=0.7114357948303223,
beta=0.3333333333333333, time=18.90371537208557
Epoch 11: train_loss=0.7115664482116699, val_loss=0.7105739116668701,
beta=0.36666666666666664, time=19.064379930496216
Epoch 12: train_loss=0.7108173966407776, val_loss=0.7099395394325256,
beta=0.4, time=19.305054903030396
Epoch 13: train_loss=0.7101265788078308, val_loss=0.709210991859436,
beta=0.43333333333333335, time=19.490837574005127
Epoch 14: train_loss=0.7097421884536743, val_loss=0.7091636061668396,
beta=0.4666666666666667, time=20.039698362350464
Epoch 15: train_loss=0.7092770338058472, val_loss=0.7080751657485962,
beta=0.5, time=20.599578857421875
Epoch 16: train_loss=0.7087876796722412, val_loss=0.7077639698982239,
beta=0.5333333333333333, time=19.763303756713867
Epoch 17: train_loss=0.7083626389503479, val_loss=0.7080010771751404,
beta=0.5666666666666667, time=20.24574112892151
Epoch 18: train_loss=0.7081132531166077, val_loss=0.7072455883026123,
beta=0.6, time=20.274344444274902
Epoch 19: train_loss=0.707838773727417, val_loss=0.7070552110671997,
beta=0.6333333333333333, time=20.3120174407959
Epoch 20: train_loss=0.7071681022644043, val_loss=0.7065160870552063,
beta=0.6666666666666666, time=19.220548152923584
Epoch 21: train_loss=0.7070402503013611, val_loss=0.7065091729164124,
beta=0.7, time=18.740514278411865
Epoch 22: train_loss=0.7066516876220703, val_loss=0.7060900926589966,
beta=0.7333333333333333, time=19.023975610733032
Epoch 23: train_loss=0.7063233852386475, val_loss=0.7052718997001648,
beta=0.7666666666666667, time=19.50731921195984
Epoch 24: train_loss=0.7061111330986023, val_loss=0.705385148525238,
beta=0.8, time=18.497273206710815
Epoch 25: train_loss=0.7058065533638, val_loss=0.7049115300178528,
beta=0.8333333333333334, time=18.53290867805481
Epoch 26: train_loss=0.7053521275520325, val_loss=0.7047088742256165,
beta=0.8666666666666667, time=18.554425954818726
Epoch 27: train_loss=0.7050511240959167, val_loss=0.7043158411979675,
beta=0.9, time=18.99888849258423
Epoch 28: train_loss=0.7047553658485413, val_loss=0.7040398120880127,
beta=0.9333333333333333, time=19.30444836616516
Epoch 29: train_loss=0.7045000791549683, val_loss=0.7037945985794067,
beta=0.9666666666666667, time=18.870641708374023
Epoch 30: train_loss=0.7040393948554993, val_loss=0.7034760117530823,
beta=0.0, time=18.9162437915802
Epoch 31: train_loss=0.7037041783332825, val_loss=0.7028787136077881,
beta=0.03333333333333333, time=17.863133192062378
Epoch 32: train_loss=0.7037039995193481, val_loss=0.7032443284988403,
beta=0.06666666666666667, time=17.77161192893982
Epoch 33: train_loss=0.7031100392341614, val_loss=0.7024551630020142,
beta=0.1, time=17.933035373687744
Epoch 34: train_loss=0.7029274702072144, val_loss=0.7022075057029724,
beta=0.13333333333333333, time=17.860363245010376
Epoch 35: train_loss=0.7027343511581421, val_loss=0.7020270824432373,
beta=0.16666666666666666, time=18.80611777305603
Epoch 36: train_loss=0.7023335695266724, val_loss=0.7014731764793396,
beta=0.2, time=18.17232632637024
Epoch 37: train_loss=0.7021134495735168, val_loss=0.7015348672866821,
beta=0.23333333333333334, time=17.778127670288086
Epoch 38: train_loss=0.7019042372703552, val_loss=0.7008528709411621,
beta=0.26666666666666666, time=18.157698392868042
```

```
Epoch 39: train_loss=0.7014980316162109, val_loss=0.700861394405365,
beta=0.3, time=17.91987633705139
Epoch 40: train_loss=0.7013727426528931, val_loss=0.700497031211853,
beta=0.3333333333333333, time=17.94876456260681
Epoch 41: train_loss=0.7010940909385681, val_loss=0.7003654837608337,
beta=0.36666666666666664, time=17.823878526687622
Epoch 42: train_loss=0.7008051872253418, val_loss=0.7003440260887146,
beta=0.4, time=17.88017177581787
Epoch 43: train_loss=0.7006179690361023, val_loss=0.7000809907913208,
beta=0.43333333333333335, time=17.795143365859985
Epoch 44: train_loss=0.7004104256629944, val_loss=0.6997069120407104,
beta=0.4666666666666667, time=17.81314992904663
Epoch 45: train_loss=0.7001534700393677, val_loss=0.6993126273155212,
beta=0.5, time=17.948390245437622
Epoch 46: train_loss=0.6997759342193604, val_loss=0.6990453600883484,
beta=0.5333333333333333, time=17.730741500854492
Epoch 47: train_loss=0.6997024416923523, val_loss=0.6988089084625244,
beta=0.5666666666666667, time=17.84877109527588
Epoch 48: train_loss=0.6995878219604492, val_loss=0.6988939642906189,
beta=0.6, time=17.782766103744507
Epoch 49: train_loss=0.6992529630661011, val_loss=0.6988014578819275,
beta=0.6333333333333333, time=17.78438425064087
Epoch 50: train_loss=0.6989504098892212, val_loss=0.6982407569885254,
beta=0.6666666666666666, time=17.71418023109436
Epoch 51: train_loss=0.6986980438232422, val_loss=0.6980570554733276,
beta=0.7, time=17.699395179748535
Epoch 52: train_loss=0.6985976099967957, val_loss=0.697930634021759,
beta=0.7333333333333333, time=17.69115447998047
Epoch 53: train_loss=0.6984084248542786, val_loss=0.6976491212844849,
beta=0.7666666666666667, time=17.614716291427612
Epoch 54: train_loss=0.6982346177101135, val_loss=0.6977491974830627,
beta=0.8, time=17.562248468399048
Epoch 55: train_loss=0.6980628967285156, val_loss=0.6971568465232849,
beta=0.8333333333333334, time=17.592220783233643
Epoch 56: train_loss=0.6977051496505737, val_loss=0.6970228552818298,
beta=0.8666666666666667, time=17.613053560256958
Epoch 57: train_loss=0.6977314352989197, val_loss=0.6971796154975891,
beta=0.9, time=17.6390540599823
Epoch 58: train_loss=0.69757080078125, val_loss=0.6964827179908752,
beta=0.9333333333333333, time=19.0875027179718
Epoch 59: train_loss=0.6971507668495178, val_loss=0.6966734528541565,
beta=0.9666666666666667, time=19.593557834625244
Epoch 60: train_loss=0.6971234083175659, val_loss=0.6963738203048706,
beta=0.0, time=18.186825037002563
Epoch 61: train_loss=0.696854829788208, val_loss=0.6990774869918823,
beta=0.03333333333333333, time=18.00300931930542
Epoch 62: train_loss=0.696959376335144, val_loss=0.6969718337059021,
beta=0.06666666666666667, time=18.593783140182495
Epoch 63: train_loss=0.6965886950492859, val_loss=0.6963696479797363,
beta=0.1, time=18.351609706878662
Epoch 64: train_loss=0.6965106129646301, val_loss=0.6958004832267761,
beta=0.13333333333333333, time=18.88807201385498
Epoch 65: train_loss=0.6963855624198914, val_loss=0.6955680847167969,
beta=0.16666666666666666, time=17.06527018547058
Epoch 66: train_loss=0.6961843967437744, val_loss=0.6956273317337036,
beta=0.2, time=16.467246294021606
Epoch 67: train_loss=0.6961728930473328, val_loss=0.6956288814544678,
beta=0.23333333333333334, time=16.38812255859375
Epoch 68: train_loss=0.6960312724113464, val_loss=0.6952754855155945,
beta=0.26666666666666666, time=16.434219360351562
Epoch 69: train_loss=0.6957453489303589, val_loss=0.6951357126235962,
beta=0.3, time=16.40023112297058
Epoch 70: train_loss=0.6957116723060608, val_loss=0.6949580907821655,
beta=0.3333333333333333, time=16.42564845085144
Epoch 71: train_loss=0.6955066323280334, val_loss=0.6950899362564087,
beta=0.36666666666666664, time=16.39795207977295
Epoch 72: train_loss=0.6953009366989136, val_loss=0.6947097182273865,
beta=0.4, time=16.50849747657776
Epoch 73: train_loss=0.6952520608901978, val_loss=0.6945493221282959,
beta=0.43333333333333335, time=16.360748052597046
Epoch 74: train_loss=0.6952635645866394, val_loss=0.6945231556892395,
beta=0.4666666666666667, time=16.32266855239868
Epoch 75: train_loss=0.6950370669364929, val_loss=0.6943386197090149,
beta=0.5, time=16.476413249969482
Epoch 76: train_loss=0.6949123740196228, val_loss=0.6941936612129211,
beta=0.5333333333333333, time=16.531242609024048
Epoch 77: train_loss=0.694852888584137, val_loss=0.694151759147644,
beta=0.5666666666666667, time=16.545607328414917
Epoch 78: train_loss=0.6946910619735718, val_loss=0.6942607760429382,
beta=0.6, time=16.44561195373535
Epoch 79: train_loss=0.6947070360183716, val_loss=0.6940546631813049,
beta=0.6333333333333333, time=16.395132064819336
Epoch 80: train_loss=0.6944797039031982, val_loss=0.693905770778656,
beta=0.6666666666666666, time=16.42884588241577
Epoch 81: train_loss=0.6943473219871521, val_loss=0.6937331557273865,
beta=0.7, time=16.67229175567627
Epoch 82: train_loss=0.6942225694656372, val_loss=0.6935571432113647,
beta=0.7333333333333333, time=16.24709725379944
Epoch 83: train_loss=0.6941530108451843, val_loss=0.6936181783676147,
beta=0.7666666666666667, time=16.321619749069214
Epoch 84: train_loss=0.6941514611244202, val_loss=0.6936118602752686,
beta=0.8, time=16.39912486076355
Epoch 85: train_loss=0.6941747665405273, val_loss=0.6933795809745789,
beta=0.8333333333333334, time=16.599560976028442
Epoch 86: train_loss=0.6939951777458191, val_loss=0.6932218074798584,
beta=0.8666666666666667, time=17.78993582725525
Epoch 87: train_loss=0.6938964128494263, val_loss=0.6932523846626282,
beta=0.9, time=17.548035383224487
```

```
Epoch 88: train_loss=0.6937994360923767, val_loss=0.6933560371398926,
beta=0.9333333333333333, time=17.601561784744263
Epoch 89: train_loss=0.6935822367668152, val_loss=0.6929031014442444,
beta=0.9666666666666667, time=18.136427640914917
Epoch 90: train_loss=0.6935515999794006, val_loss=0.7665694355964661,
beta=0.0, time=18.56167435646057
Epoch 91: train_loss=0.6936132311820984, val_loss=0.6929948329925537,
beta=0.03333333333333333, time=18.078909397125244
Epoch 92: train_loss=0.6934200525283813, val_loss=0.6929394006729126,
beta=0.06666666666666667, time=17.898712396621704
Epoch 93: train_loss=0.6933131217956543, val_loss=0.6929594874382019,
beta=0.1, time=17.60557770729065
Epoch 94: train_loss=0.6932167410850525, val_loss=0.6926475167274475,
beta=0.13333333333333333, time=17.733420372009277
Epoch 95: train_loss=0.6931201815605164, val_loss=0.692618191242218,
beta=0.16666666666666666, time=17.98951506614685
Epoch 96: train_loss=0.6931238174438477, val_loss=0.6924722790718079,
beta=0.2, time=18.018215894699097
Epoch 97: train_loss=0.6930025219917297, val_loss=0.6926053762435913,
beta=0.23333333333333334, time=17.777899742126465
Epoch 98: train_loss=0.6929662823677063, val_loss=0.6926819086074829,
beta=0.26666666666666666, time=17.92274308204651
Epoch 99: train_loss=0.6929201483726501, val_loss=0.6922724843025208,
beta=0.3, time=17.800713300704956
Epoch 100: train_loss=0.6928223967552185, val_loss=0.6923108696937561,
beta=0.3333333333333333, time=17.651853322982788
Epoch 101: train_loss=0.6927914619445801, val_loss=0.6922712922096252,
beta=0.36666666666666664, time=18.745437622070312
Epoch 102: train_loss=0.692696213722229, val_loss=0.692059338092804,
beta=0.4, time=18.283118963241577
Epoch 103: train_loss=0.6926394701004028, val_loss=0.692043125629425,
beta=0.43333333333333335, time=17.945343494415283
Epoch 104: train_loss=0.6925063729286194, val_loss=0.6921292543411255,
beta=0.4666666666666667, time=17.809961318969727
Epoch 105: train_loss=0.6925274133682251, val_loss=0.6919190287590027,
beta=0.5, time=17.77335262298584
Epoch 106: train_loss=0.6925604939460754, val_loss=0.6919451951980591,
beta=0.5333333333333333, time=17.853723526000977
Epoch 107: train_loss=0.6923950910568237, val_loss=0.6919437050819397,
beta=0.5666666666666667, time=18.353167295455933
Epoch 108: train_loss=0.6923657059669495, val_loss=0.6916803121566772,
beta=0.6, time=18.16617465019226
Epoch 109: train_loss=0.6923002004623413, val_loss=0.6917513012886047,
beta=0.6333333333333333, time=18.153886318206787
Epoch 110: train_loss=0.6921660900115967, val_loss=0.6915661692619324,
beta=0.6666666666666666, time=17.887900590896606
Epoch 111: train_loss=0.6922193169593811, val_loss=0.6916059851646423,
beta=0.7, time=18.151943922042847
Epoch 112: train_loss=0.6920351982116699, val_loss=0.6916785836219788,
beta=0.7333333333333333, time=17.632816791534424
Epoch 113: train_loss=0.6920760273933411, val_loss=0.6913166642189026,
beta=0.7666666666666667, time=18.24550223350525
Epoch 114: train_loss=0.6919795274734497, val_loss=0.6914474964141846,
beta=0.8, time=17.685124397277832
Epoch 115: train_loss=0.6918575167655945, val_loss=0.691393256187439,
beta=0.8333333333333334, time=17.948457717895508
Epoch 116: train_loss=0.6918394565582275, val_loss=0.6914357542991638,
beta=0.8666666666666667, time=18.496946334838867
Epoch 117: train_loss=0.6918550729751587, val_loss=0.691301226615905,
beta=0.9, time=18.013478755950928
Epoch 118: train_loss=0.6918106079101562, val_loss=0.6912075281143188,
beta=0.9333333333333333, time=18.08651065826416
Epoch 119: train_loss=0.6917456388473511, val_loss=0.691184937953949,
beta=0.9666666666666667, time=18.133899211883545
Epoch 120: train_loss=0.6916614174842834, val_loss=0.6912009119987488,
beta=0.0, time=17.722911834716797
```

IV Test modelu

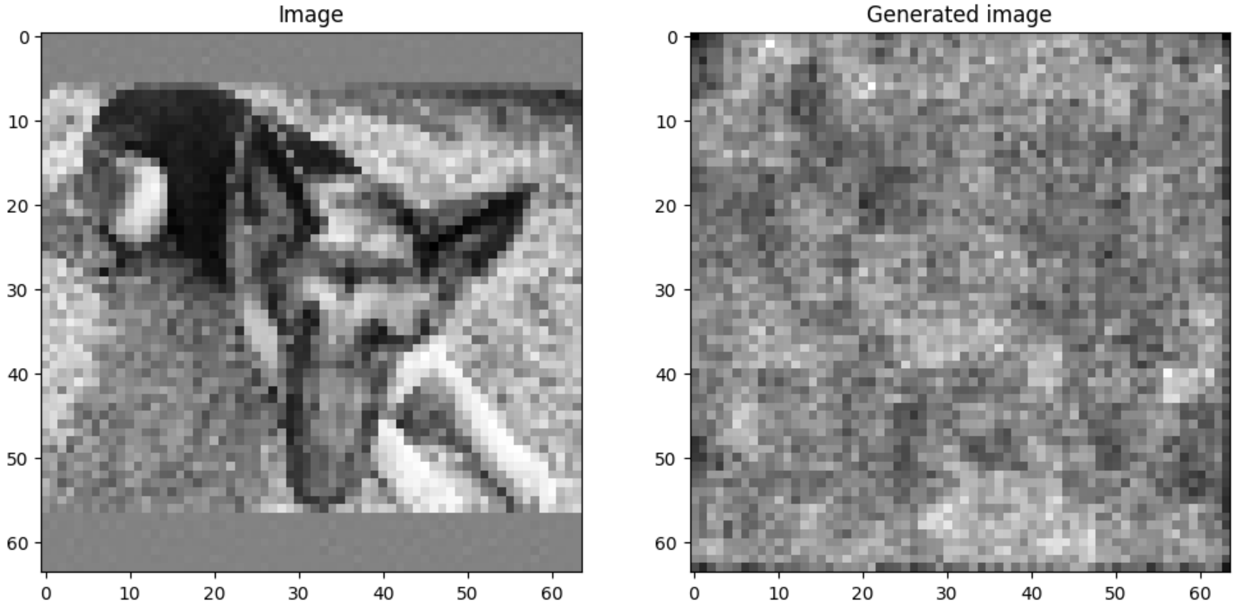
```python
model.load('best')

test_loss = tf.keras.metrics.Mean()
for test_x in test_dataset:
    test_loss(compute_loss(model, test_x)[0])
print(f'test_loss={test_loss.result()}')
```

```
test_loss=0.6904774308204651
```

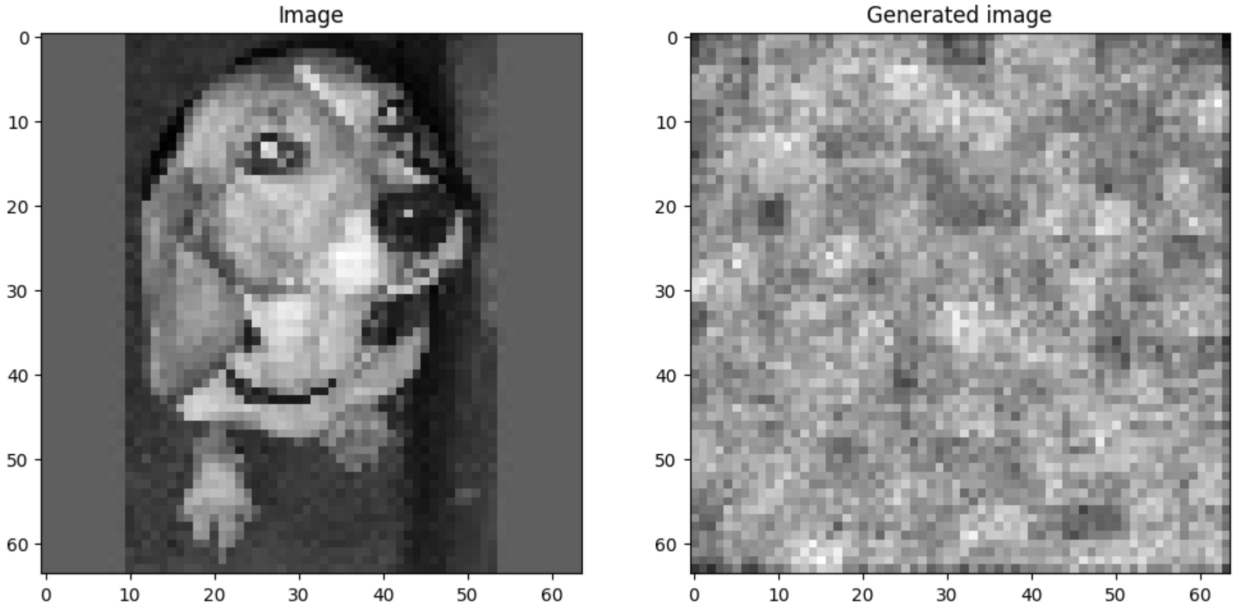
```python
import matplotlib.pyplot as plt

def compare_images(image, generated_image):

    fig, ax = plt.subplots(1,2, figsize=(12,6))

    ax[0].imshow(image, cmap='gray')
    ax[0].set_title('Image')
    ax[1].imshow(generated_image, cmap='gray')
    ax[1].set_title('Generated image')

    plt.show()

def generate_image(image):
    encoded = model.encoder(image.reshape((1, 64, 64, 1)))
    args = tf.split(encoded, num_or_size_splits=2, axis=1)
    encoded = latent_reparam_sampling(args)

    return model.decoder.predict(encoded)[0]
```

```python
sample_size = 2
sample = np.random.randint(0, X_test.shape[0], sample_size)
for i in range(sample_size):
    decoded = generate_image(X_test[sample[i]])
    denormalized_image = X_test[sample[i]] * 255.0
    compare_images(denormalized_image, decoded)
```

```
1/1 [==============================] - 0s 152ms/step
```

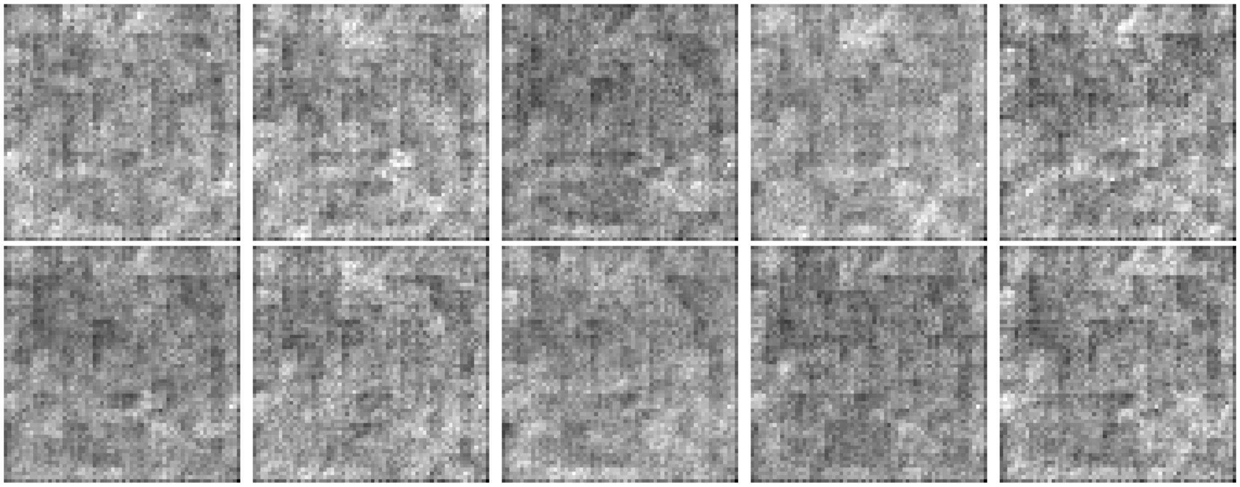


```
1/1 [==============================] - 0s 18ms/step
```



V Generowanie 10 obrazów

```
seed = 33
random_representation = tf.random.uniform((10, 1024), seed=seed)
images = model.decoder(random_representation).numpy().reshape(10, 64,
64, 1)

fig, axes = plt.subplots(nrows=2, ncols=5, figsize=(15, 6))

for i, image in enumerate(images):
    ax = axes[i // 5, i % 5]
    ax.imshow(image.reshape(64, 64, 1), cmap='gray')
    ax.axis('off')

plt.tight_layout()
plt.show()

plt.show()
```



Additional testing

```
extra_test_dirpath = 'test-dirpath/'
filenames = [filename for filename in os.listdir(extra_test_dirpath)]
print(f"Total number of images: {len(filenames)}")

extra_test_preprocessed_dirpath = 'extra-test-dog-preprocessed/'

if not os.path.exists(extra_test_preprocessed_dirpath):
    os.makedirs(extra_test_preprocessed_dirpath)

start_time = time.time()
for filename in filenames:
    try:
        process_image(filename, extra_test_dirpath,
extra_test_preprocessed_dirpath)
    except Exception as e:
        print(f"An error occurred: {e}")

end_time = time.time()
print(f"Total time: {end_time - start_time}")

X_test_extra = np.zeros((len(filenames), IMAGE_HEIGHT, IMAGE_WIDTH,
1), dtype=np.uint8)
for index, filename in enumerate(filenames):
    image = cv2.imread(extra_test_preprocessed_dirpath + filename,
cv2.IMREAD_GRAYSCALE)
    X_test_extra[index] = image.reshape(IMAGE_HEIGHT, IMAGE_WIDTH, 1)

X_test_extra = X_test_extra / 255.0

# X_test_extra = X_train[:10]
seed = 33
extra_test_sample_count = 10
images = model.decoder(random_representation).numpy().reshape(10, 64,
64, 1)

fig, axes = plt.subplots(nrows=2, ncols=5, figsize=(15, 6))

for i in range(extra_test_sample_count):
    image = generate_image(X_test_extra[i])
    ax = axes[i // 5, i % 5]
    ax.imshow(image.reshape(64, 64, 1), cmap='gray')
    ax.axis('off')

plt.tight_layout()
plt.show()

plt.show()
```

```
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 20ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 39ms/step
1/1 [==============================] - 0s 37ms/step
1/1 [==============================] - 0s 19ms/step
1/1 [==============================] - 0s 17ms/step
1/1 [==============================] - 0s 18ms/step
1/1 [==============================] - 0s 18ms/step
```