

▼ Dicoding Indonesia

Belajar Machine Learning untuk Pemula

This file is a submission for the Final Project of [Belajar Machine Learning untuk Pemula](#)

Made by = [Muhammad Surya Sanjiwani](#)

▼ Showing the running time of each cell

```
!pip install ipython-autotime
%load_ext autotime
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting ipython-autotime
  Downloading ipython-autotime-0.3.1-py2.py3-none-any.whl (6.8 kB)
Requirement already satisfied: ipython in /usr/local/lib/python3.10/dist-packages (from ipython-autotime) (7.34.0)
Requirement already satisfied: setuptools>=18.5 in /usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (67.7.2)
Collecting jedi>=0.16 (from ipython->ipython-autotime)
  Downloading jedi-0.18.2-py2.py3-none-any.whl (1.6 MB)
    1.6/1.6 MB 60.8 MB/s eta 0:00:00
Requirement already satisfied: decorator in /usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (4.4.2)
Requirement already satisfied: pickleshare in /usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (0.7.5)
Requirement already satisfied: traitlets>=4.2 in /usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (5.7.1)
Requirement already satisfied: prompt-toolkit!=3.0.0,!<3.0.1,<3.1.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from ipytho
Requirement already satisfied: pygments in /usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (2.14.0)
Requirement already satisfied: backcall in /usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (0.2.0)
Requirement already satisfied: matplotlib-inline in /usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (0.1.6)
Requirement already satisfied: pexpect>4.3 in /usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (4.8.0)
Requirement already satisfied: parso<0.9.0,>=0.8.0 in /usr/local/lib/python3.10/dist-packages (from jedi>=0.16->ipython->ipython-au
Requirement already satisfied: ptyprocess>=0.5 in /usr/local/lib/python3.10/dist-packages (from pexpect>4.3->ipython->ipython-autot
Requirement already satisfied: wcwidth in /usr/local/lib/python3.10/dist-packages (from prompt-toolkit!=3.0.0,!<3.0.1,<3.1.0,>=2.0.
Installing collected packages: jedi, ipython-autotime
Successfully installed ipython-autotime-0.3.1 jedi-0.18.2
time: 249 µs (started: 2023-06-20 14:15:49 +00:00)
```

▼ Import dataset with wget

```
!wget http://github.com/dicodingacademy/assets/releases/download/release/rockpaperscissors.zip

--2023-06-20 14:17:00-- http://github.com/dicodingacademy/assets/releases/download/release/rockpaperscissors.zip
Resolving github.com (github.com)... 140.82.114.4
Connecting to github.com (github.com)|140.82.114.4|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://github.com/dicodingacademy/assets/releases/download/release/rockpaperscissors.zip [following]
--2023-06-20 14:17:00-- https://github.com/dicodingacademy/assets/releases/download/release/rockpaperscissors.zip
Connecting to github.com (github.com)|140.82.114.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/391417272/7eb836f2-695b-4a46-9c78-b658671669
--2023-06-20 14:17:00-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/391417272/7eb836f2-695b-4a46-
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.111.133, 185.199.108.133, 185.199.110.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.111.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 322873683 (308M) [application/octet-stream]
Saving to: 'rockpaperscissors.zip'

rockpaperscissors.z 100%[=====] 307.92M 6.65MB/s in 5.2s

2023-06-20 14:17:06 (59.7 MB/s) - 'rockpaperscissors.zip' saved [322873683/322873683]

time: 6.23 s (started: 2023-06-20 14:17:00 +00:00)
```

▼ Unzip the dataset

```
!unzip /content/rockpaperscissors.zip
```

```

inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._otBEUIQNFGMk80vF.png
inflating: rockpapersciissors/rps-cv-images/scissors/yiOfEmSzPhjaYDwH.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._yiOfEmSzPhjaYDwH.png
inflating: rockpapersciissors/rps-cv-images/scissors/ZcGYyQZWKP4EcVEz.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._ZcGYyQZWKP4EcVEz.png
inflating: rockpapersciissors/rps-cv-images/scissors/NRzk5UfsKrSzt2j.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._NRzk5UfsKrSzt2j.png
inflating: rockpapersciissors/rps-cv-images/scissors/tf8qhgtb5y3S8UiF.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._tf8qhgtb5y3S8UiF.png
inflating: rockpapersciissors/rps-cv-images/scissors/JULz0bpEBDkoMfh4.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._JULz0bpEBDkoMfh4.png
inflating: rockpapersciissors/rps-cv-images/scissors/277q9TUTAsBABTCj.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._277q9TUTAsBABTCj.png
inflating: rockpapersciissors/rps-cv-images/scissors/56VICUKU6RgoLTHW.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._56VICUKU6RgoLTHW.png
inflating: rockpapersciissors/rps-cv-images/scissors/EWWFVEzkNulhftg2.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._EWWFVEzkNulhftg2.png
inflating: rockpapersciissors/rps-cv-images/scissors/KM8TqJbWqew79vPt.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._KM8TqJbWqew79vPt.png
inflating: rockpapersciissors/rps-cv-images/scissors/4burI8a7jyfTRB16.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._4burI8a7jyfTRB16.png
inflating: rockpapersciissors/rps-cv-images/scissors/hP1nFePk6ao9xKDF.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._hP1nFePk6ao9xKDF.png
inflating: rockpapersciissors/rps-cv-images/scissors/xgFgqnPJlU3u1ZqY.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._xgFgqnPJlU3u1ZqY.png
inflating: rockpapersciissors/rps-cv-images/scissors/2vDaPrc35RGC8nvM.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._2vDaPrc35RGC8nvM.png
inflating: rockpapersciissors/rps-cv-images/scissors/QUqw1jotp4m0m1ZY.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._QUqw1jotp4m0m1ZY.png
inflating: rockpapersciissors/rps-cv-images/scissors/m5ykDPqbTkPOEgWw.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._m5ykDPqbTkPOEgWw.png
inflating: rockpapersciissors/rps-cv-images/scissors/ny4ARFjALtDxJqGg.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._ny4ARFjALtDxJqGg.png
inflating: rockpapersciissors/rps-cv-images/scissors/MFG70ohG4iZ3xm3b.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._MFG70ohG4iZ3xm3b.png
inflating: rockpapersciissors/rps-cv-images/scissors/jL3CMVgsLdWznnKe.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._jL3CMVgsLdWznnKe.png
inflating: rockpapersciissors/rps-cv-images/scissors/jy6fSFQ1ynecl3P7.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._jy6fSFQ1ynecl3P7.png
inflating: rockpapersciissors/rps-cv-images/scissors/bAQiT59XVn1Kqc4.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._bAQiT59XVn1Kqc4.png
inflating: rockpapersciissors/rps-cv-images/scissors/OLq5gEcrMDGftxeG.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._OLq5gEcrMDGftxeG.png
inflating: rockpapersciissors/rps-cv-images/scissors/mRlkMzgNq3PxHHe4.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._mRlkMzgNq3PxHHe4.png
inflating: rockpapersciissors/rps-cv-images/scissors/Gr3EpS7A03KtWTz0.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._Gr3EpS7A03KtWTz0.png
inflating: rockpapersciissors/rps-cv-images/scissors/fYsFOMTEQ0VIy1Ma.png
inflating: __MACOSX/rockpapersciissors/rps-cv-images/scissors/._fYsFOMTEQ0VIy1Ma.png

```

▼ Define folder and variable

```

import os
f_gunting = os.path.join('/content/rockpapersciissors/scissors')
f_batu = os.path.join('/content/rockpapersciissors/rock')
f_kertas = os.path.join('/content/rockpapersciissors/paper')

time: 566 µs (started: 2023-06-20 14:17:41 +00:00)

```

▼ Import module

```

import tensorflow as tf
import keras.preprocessing
from keras.preprocessing import image
from keras.preprocessing.image import ImageDataGenerator

time: 1.07 ms (started: 2023-06-20 15:40:25 +00:00)

```

▼ Augment dataset picture to increase variety of dataset

```

f_main = "/content/rockpapersciissors/rps-cv-images/"
augmentasi = ImageDataGenerator(
    rescale = 1./255,
    rotation_range=35,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    fill_mode='nearest',

```

```
validation_split=0.4
)
```

time: 1.29 ms (started: 2023-06-20 15:40:36 +00:00)

time: 15.2 ms (started: 2022-12-19 10:38:28 +00:00)

▼ Divide the dataset into data train and data validate

```
data_train = augmentasi.flow_from_directory(
    f_main,
    target_size=(150,150),
    class_mode='categorical',
    shuffle=True,
    batch_size=50,
    subset='training'
)
```

```
data_validate = augmentasi.flow_from_directory(
    f_main,
    target_size=(150,150),
    class_mode='categorical',
    shuffle=True,
    batch_size=50,
    subset='validation'
)
```

Found 1314 images belonging to 3 classes.
Found 874 images belonging to 3 classes.
time: 82.8 ms (started: 2023-06-20 15:41:10 +00:00)

▼ Defines a convolutional neural network (CNN) using the TensorFlow Keras API

```
model = tf.keras.models.Sequential([

    tf.keras.layers.Conv2D(16, (3,3), activation='relu', input_shape=(150, 150, 3)),
    tf.keras.layers.MaxPooling2D(2, 2),

    tf.keras.layers.Conv2D(32, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),

    tf.keras.layers.Conv2D(32, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),

    tf.keras.layers.Conv2D(64, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),

    tf.keras.layers.Conv2D(64, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),

    tf.keras.layers.Flatten(),
    tf.keras.layers.Dropout(0.5),

    tf.keras.layers.Dense(128, activation='relu'),

    tf.keras.layers.Dense(512, activation='relu'),

    tf.keras.layers.Dense(3, activation='softmax')
])
```

```
model.summary()
```

Model: "sequential"

| Layer (type) | Output Shape | Param # |
|--------------------------------|----------------------|---------|
| ===== | | |
| conv2d (Conv2D) | (None, 148, 148, 16) | 448 |
| max_pooling2d (MaxPooling2D) | (None, 74, 74, 16) | 0 |
| conv2d_1 (Conv2D) | (None, 72, 72, 32) | 4640 |
| max_pooling2d_1 (MaxPooling2D) | (None, 36, 36, 32) | 0 |
| conv2d_2 (Conv2D) | (None, 34, 34, 32) | 9248 |

```
max_pooling2d_2 (MaxPooling (None, 17, 17, 32)      0
2D)

conv2d_3 (Conv2D)          (None, 15, 15, 64)      18496

max_pooling2d_3 (MaxPooling (None, 7, 7, 64)      0
2D)

conv2d_4 (Conv2D)          (None, 5, 5, 64)       36928

max_pooling2d_4 (MaxPooling (None, 2, 2, 64)      0
2D)

flatten (Flatten)         (None, 256)            0

dropout (Dropout)         (None, 256)            0

dense (Dense)             (None, 128)           32896

dense_1 (Dense)           (None, 512)          66048

dense_2 (Dense)           (None, 3)             1539

=====
Total params: 170,243
Trainable params: 170,243
Non-trainable params: 0
time: 383 ms (started: 2023-06-20 15:41:13 +00:00)
```

▼ Compile the model

```
model.compile(loss = 'categorical_crossentropy',
              optimizer='rmsprop',
              metrics=['accuracy'])

time: 16.4 ms (started: 2023-06-20 15:41:17 +00:00)
```

▼ Training model

```
epoch = 25
history = model.fit(
    data_train,
    steps_per_epoch=25,
    epochs=epoch,
    validation_data=data_validate,
    validation_steps=4,
    verbose=2
)

print('finish training')
```

Epoch 1/25
25/25 - 29s - loss: 0.1953 - accuracy: 0.9382 - val_loss: 0.1381 - val_accuracy: 0.9400 - 29s/epoch - 1s/step
Epoch 2/25
25/25 - 24s - loss: 0.1989 - accuracy: 0.9399 - val_loss: 0.1330 - val_accuracy: 0.9800 - 24s/epoch - 971ms/step
Epoch 3/25
25/25 - 23s - loss: 0.1848 - accuracy: 0.9374 - val_loss: 0.1022 - val_accuracy: 0.9700 - 23s/epoch - 930ms/step
Epoch 4/25
25/25 - 24s - loss: 0.1903 - accuracy: 0.9325 - val_loss: 0.2189 - val_accuracy: 0.9350 - 24s/epoch - 961ms/step
Epoch 5/25
25/25 - 23s - loss: 0.1502 - accuracy: 0.9555 - val_loss: 0.1248 - val_accuracy: 0.9450 - 23s/epoch - 932ms/step
Epoch 6/25
25/25 - 23s - loss: 0.1443 - accuracy: 0.9539 - val_loss: 0.1026 - val_accuracy: 0.9750 - 23s/epoch - 924ms/step
Epoch 7/25
25/25 - 24s - loss: 0.1540 - accuracy: 0.9514 - val_loss: 0.1558 - val_accuracy: 0.9500 - 24s/epoch - 962ms/step
Epoch 8/25
25/25 - 24s - loss: 0.1863 - accuracy: 0.9448 - val_loss: 0.0868 - val_accuracy: 0.9850 - 24s/epoch - 959ms/step
Epoch 9/25
25/25 - 24s - loss: 0.1357 - accuracy: 0.9580 - val_loss: 0.2318 - val_accuracy: 0.9400 - 24s/epoch - 956ms/step
Epoch 10/25
25/25 - 24s - loss: 0.1638 - accuracy: 0.9448 - val_loss: 0.1509 - val_accuracy: 0.9500 - 24s/epoch - 960ms/step
Epoch 11/25
25/25 - 23s - loss: 0.1042 - accuracy: 0.9687 - val_loss: 0.1059 - val_accuracy: 0.9700 - 23s/epoch - 938ms/step
Epoch 12/25
25/25 - 24s - loss: 0.1428 - accuracy: 0.9629 - val_loss: 0.0394 - val_accuracy: 0.9850 - 24s/epoch - 957ms/step
Epoch 13/25
25/25 - 24s - loss: 0.1310 - accuracy: 0.9654 - val_loss: 0.5550 - val_accuracy: 0.8000 - 24s/epoch - 954ms/step
Epoch 14/25
25/25 - 24s - loss: 0.1075 - accuracy: 0.9629 - val_loss: 0.0642 - val_accuracy: 0.9850 - 24s/epoch - 957ms/step
Epoch 15/25
25/25 - 24s - loss: 0.1136 - accuracy: 0.9563 - val_loss: 0.1811 - val_accuracy: 0.9650 - 24s/epoch - 975ms/step

```
Epoch 16/25
25/25 - 24s - loss: 0.1090 - accuracy: 0.9588 - val_loss: 0.1033 - val_accuracy: 0.9750 - 24s/epoch - 962ms/step
Epoch 17/25
25/25 - 24s - loss: 0.1190 - accuracy: 0.9621 - val_loss: 0.1842 - val_accuracy: 0.9650 - 24s/epoch - 958ms/step
Epoch 18/25
25/25 - 24s - loss: 0.1368 - accuracy: 0.9555 - val_loss: 0.1429 - val_accuracy: 0.9650 - 24s/epoch - 970ms/step
Epoch 19/25
25/25 - 23s - loss: 0.1471 - accuracy: 0.9555 - val_loss: 0.0679 - val_accuracy: 0.9850 - 23s/epoch - 931ms/step
Epoch 20/25
25/25 - 24s - loss: 0.0886 - accuracy: 0.9768 - val_loss: 0.1778 - val_accuracy: 0.9550 - 24s/epoch - 978ms/step
Epoch 21/25
25/25 - 26s - loss: 0.1309 - accuracy: 0.9588 - val_loss: 0.1163 - val_accuracy: 0.9700 - 26s/epoch - 1s/step
Epoch 22/25
25/25 - 23s - loss: 0.0997 - accuracy: 0.9662 - val_loss: 0.0496 - val_accuracy: 0.9850 - 23s/epoch - 931ms/step
Epoch 23/25
25/25 - 24s - loss: 0.1402 - accuracy: 0.9621 - val_loss: 0.0418 - val_accuracy: 0.9800 - 24s/epoch - 956ms/step
Epoch 24/25
25/25 - 24s - loss: 0.1241 - accuracy: 0.9621 - val_loss: 0.0665 - val_accuracy: 0.9850 - 24s/epoch - 957ms/step
Epoch 25/25
25/25 - 23s - loss: 0.1100 - accuracy: 0.9712 - val_loss: 0.1173 - val_accuracy: 0.9550 - 23s/epoch - 922ms/step
finish training
time: 13min 3s (started: 2023-06-20 17:32:38 +00:00)
```

▼ Checking the accuracy

```
import statistics
accuracy = (statistics.mean(history.history['val_accuracy']))*100
print('The accuracy is :',round(accuracy,3),'%')

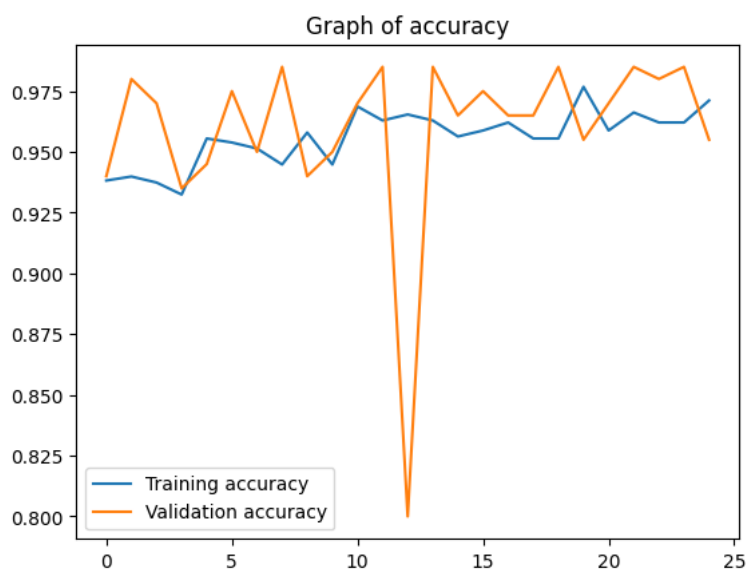
The accuracy is : 95.98 %
time: 2.62 ms (started: 2023-06-20 17:48:18 +00:00)
```

▼ Graph of accuracy from training and validating

```
import matplotlib.pyplot as plt
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(len(acc))

plt.plot(epochs, acc, label='Training accuracy ')
plt.plot(epochs, val_acc, label='Validation accuracy')
plt.title('Graph of accuracy')
plt.legend(loc=0)
plt.figure()
plt.show()
```



<Figure size 640x480 with 0 Axes>
time: 205 ms (started: 2023-06-20 17:48:26 +00:00)

▼ Import local image to test the model bold text

```
import numpy as np
```

```

from google.colab import files
from tensorflow.keras.preprocessing import image
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
%matplotlib inline

```

```

uploaded = files.upload()

```

```

for fn in uploaded.keys():

```

```

    path = fn
    img = image.load_img(path, target_size=(150,150))

```

```

    imgplot = plt.imshow(img)
    x = image.img_to_array(img)
    x = np.expand_dims(x, axis=0)
    images = np.vstack([x])

```

```

    classes = model.predict(images, batch_size=10)

```

```

    print(fn)

```

```

    if classes[0][0]==1:

```

```

        print("It's paper")

```

```

    elif classes[0][1]==1:

```

```

        print("It's rock")

```

```

    elif classes[0][2]==1:

```

```

        print("It's scissor")

```

```

    else:

```

```

        print('Unknown')

```

Choose Files kertas.png

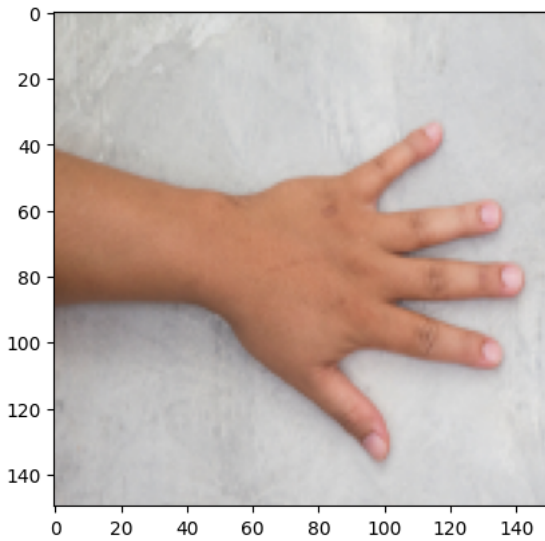
- **kertas.png**(image/png) - 621934 bytes, last modified: 6/20/2023 - 100% done

Saving kertas.png to kertas (1).png

1/1 [=====] - 0s 20ms/step

kertas.png

It's paper



time: 18.1 s (started: 2023-06-20 17:49:47 +00:00)